uc3m | Universidad Carlos III de Madrid

Grado Universitario en Ingeniería en Tecnologías de Telecomunicación

2017-2018

*Trabajo Fin de Grado*

# "Herramienta web para el análisis de ofertas de empleo"

---


Ángela Moreno Martínez

Tutora

Vanessa Gómez Verdejo


Leganés, 2018

## RESUMEN

Este Trabajo de Fin de Grado consiste en la implementación de una herramienta web para el análisis de ofertas de empleo de dos conocidos portales de búsqueda de empleo como son InfoJobs y Tecnoempleo.

Se trata de una herramienta web ya que utiliza datos de entrada, los procesa y devuelve a la salida en forma de gráficos que facilitan la comunicación con los usuarios. Estos gráficos son representaciones de distintos de los atributos que conforman las ofertas de empleo, como son su categoría o salario.

Dicha herramienta permite una interacción directa entre el usuario y su interfaz, de forma que la visualización de los diferentes gráficos varía en función de las acciones del usuario en la herramienta.

Con el fin de ofrecer una experiencia de análisis útil y eficiente, han sido considerados aquellos criterios de ofertas de empleo relevantes que permitieran sacar conclusiones precisas y descriptivas. Al tratar con grandes cantidades de datos se requiere de su tratamiento y procesamiento, así como el uso de la librería de JavaScript D3.js para poder expresarlos de forma gráfica y mostrarlos en tiempo real.

El principal objetivo del presente trabajo es facilitar al usuario una herramienta que permita, de manera visual e interactiva, obtener conclusiones en el análisis de ofertas de empleo según distintas guías.



Palabras clave:

D3, dc.js, ofertas de empleo, visualización de información, PLN, minería de datos.

*"Muere lentamente*

*quien no cambia la vida cuando está insatisfecho con su*

*trabajo, o su amor*

*Quien no arriesga lo seguro por lo incierto*

*para ir tras de un sueño*

*Quien no se permite,*

*por lo menos una vez en la vida,*

*huir de los consejos sensatos."*

Pablo Neruda

## AGRADECIMIENTOS

A mis padres, Rosa y Félix, por hacer todo esto posible de principio a fin. Por el apoyo incondicional y la paciencia. Gracias por enseñarme que la mejor recompensa se consigue tras mucho esfuerzo, siendo constante incluso en la labor por nunca rendirse.

A mis amigas y compañeros de las prácticas en empresa, por el interés y acompañamiento, por haber estado ahí en todo este proceso y haber sido testigos de mi aprendizaje.

A todos aquellos que se tomaron unos minutos para analizar mi trabajo, darme consejos constructivos y hacerme crecer.

Y, principalmente, a mi tutora, Vanessa Gómez, por toda la orientación y ayuda durante el proceso. Gracias por encontrar siempre tiempo para resolver mis dudas y hacerlo siempre con una sonrisa. Ha sido un placer.

# Índice de contenido

# Índice de tablas

# Índice de figuras

# 1. INTRODUCCIÓN

En este primer capítulo serán descritos aquellos problemas existentes que buscan ser resueltos, así como los objetivos que para ello debían cubrirse y que han motivado el desarrollo del presente Trabajo Fin de Grado. También será detallada la estructura del proyecto.

## 1.1 Motivación del trabajo.

La situación actual del mercado laboral conduce, en ocasiones, a situaciones de estrés o frustración por no encontrar un puesto de trabajo que se adapte a los requerimientos de las personas. Por ello, en los últimos años se observa una creciente competencia donde cada persona pretende buscar empleo de la forma más eficiente posible.

Aquellas personas que se encuentren en búsqueda activa de empleo pueden comprobar la creciente variabilidad de ofertas de empleo que existen actualmente en el mercado. Consecuencia de las últimas reformas laborales han surgido nuevas formas de contratación y se han reducido los costes de contratación de las empresas. La combinación de los anteriores con otros cambios ha derivado en que, aunque el número de ofertas de empleo se incremente (entre un 25% y 30% [1]), el ritmo de crecimiento interno decrezca. Por ello, es más accesible encontrar empleo, pero menos promocionar en él. Esto es motivo suficiente para que, aquel interesado en encontrar empleo, tenga la "presión" de acertar en la empresa elegida, donde pueda tener probabilidad de promocionar o desarrollar una digna carrera profesional.

Por y para satisfacer esta necesidad, no sólo la competencia ha aumentado sino también la cantidad de portales web de empleo, donde una gran cantidad de ofertas de empleo son añadidas y/o actualizadas diariamente. Estos portales permiten al usuario el filtrado de dichas ofertas en función de criterios tales como: categoría, ubicación, tipo de contrato o jornada, entre otros.

Basándome en mi propia experiencia, la búsqueda empleo acaba siendo un proceso poco emocionante, donde es difícil sacar conclusiones sobre las exigencias o tendencias del mercado y uno se rinde, tendiendo a dejar su currículum indistintamente a aquellas empresas que considera de un sector concreto y, perdiendo así la oportunidad de concentrarse más en sus objetivos al solicitar empleo, siendo ésta una de las tareas más importantes que determinan el futuro de la persona.

Así, de la necesidad de un proceso de búsqueda de empleo más atractivo, surge este trabajo de fin de grado, asumiendo la importancia de que la información que aportan los datos es mayor en caso de que dicha información se muestre gráficamente.

## 1.2 Objetivos

El objetivo de la presente herramienta de visualización es facilitar el análisis de ofertas de empleo, tratando con una cantidad considerable de ellas para poder obtener conclusiones significativas. Así, una persona podrá, visualizando las diferentes gráficas proporcionadas por la herramienta, encontrar las ofertas que se corresponden con sus requerimientos o cuál es la probabilidad de encontrar un empleo concreto según uno o varios criterios.

Para alcanzar dicho propósito, fueron planteados objetivos parciales, que serán expuestos a continuación:

- Proporcionar al usuario una interfaz interactiva con la que poder analizar ofertas de empleo provenientes de diversos portales web.
- Presentar una interfaz con diseño adaptable para una correcta visualización y utilización desde cualquier navegador utilizado.
- Incluir la posibilidad de elegir con qué datos y de qué portal de empleo se desea realizar el análisis.
- Ofrecer la máxima facilidad de modificación del código a posibles y posteriores desarrolladores.
- Realizar el pre-procesamiento de datos adecuado para una mayor significación de la información aportada por la herramienta de análisis.

## 1.3. Metodología

Para poder conseguir los anteriores objetivos, es necesario definir una metodología que incluya procesos eficientes y realistas, divididos a su vez en subprocesos o tareas que lleven a conseguir el objetivo final.

La metodología seguida a lo largo del proceso de creación de la herramienta de visualización fue la siguiente:

- En primer lugar, se realizó la documentación de las diferentes herramientas que se utilizarían posteriormente durante el desarrollo.
- Reuniones semanales con la tutora para la resolución de dudas y/o problemas en el desarrollo de la herramienta (realimentación semanal).
- Trabajo semanal en el desarrollo de la herramienta.
- Evaluación de la herramienta por terceros y las correspondientes mejoras de la herramienta.

### 1.4. Marco regulador

En este proyecto no se recogen datos del usuario, por lo cual no es pertinente analizar ninguna regulación al respecto. Sin embargo, se utilizan datos (legítimos y de uso autorizado) para el desarrollo de la herramienta presentada en esta memoria. La explotación de dichos datos se encuentra bajo responsabilidad profesional, donde su representación es fiel a la realidad y se utilizan éticamente.

### 1.5. Estructura del documento

Con el objetivo de desarrollar de forma clara y organizada el proceso llevado a cabo para obtener la herramienta final de visualización, la memoria sigue la siguiente estructura:

1. *Introducción*: en este capítulo se presenta brevemente el trabajo y se exponen los motivos que llevaron a su desarrollo, así como los objetivos que se pretenden alcanzar con ello y el marco regulador bajo el cual se encuentra. La visión global del trabajo queda enmarcada en la presente estructura y también se presenta la metodología llevada a cabo durante todo el proceso.
2. *Estado del arte*: se presentan otras herramientas que existen actualmente y que realizan funcionalidades parecidas a las que presenta este trabajo de fin de grado o TFG. También se desarrolla en este apartado una crítica al estado del arte, útil para extraer necesidades o mejoras para la realización del presente proyecto.
3. *Análisis del problema:* se examinan cuáles son las necesidades que se han de satisfacer para conseguir los objetivos propuestos en anteriores apartados.
4. *Diseño de la solución:* contiene las herramientas utilizadas para implementar la solución a los problemas expuestos en el apartado "Análisis del problema". Además, se desarrollan con detalle cada una de las fases del proyecto.
5. *Resultados*: se presenta el resultado final de la herramienta de forma que sea deducible si se han conseguido los objetivos propuestos al principio de la memoria.
6. *Evaluación:* contiene los resultados experimentales tras la prueba de la herramienta realizada por un grupo de usuarios.
7. *Planificación y presupuesto:* se presenta el proceso de planificación seguido durante la creación de este TFG y los costes que ha supuesto, así como el entorno socio-económico.
8. *Conclusiones y trabajo futuro*: se compone de todas aquellas líneas futuras en las que se podría seguir trabajando.

# 2. ESTADO DEL ARTE

## 2.1. Introducción

En este capítulo se presentarán otras herramientas que actualmente ya existen o han existido en el mercado cuya funcionalidad se podría parecer a la expuesta en el presente TFG. Previo a estas otras aplicaciones se expone en qué consiste el proceso de minería de datos, utilizado en el presente proyecto.

## 2.2. Minería de datos

La minería de datos es el proceso de transformar datos en conocimiento. Una de las definiciones enuncia que la minería de datos es la exploración de forma automática o semiautomática de grandes cantidades de datos para el descubrimiento de reglas y patrones [2].

Una etapa necesaria antes de aplicar minería de datos es el pre-procesamiento de datos, en la cual se manipulan y transforman los datos de forma que la información que contienen sea más accesible y coherente [3]. Dentro de esta transformación, se considera necesaria la limpieza de aquellos datos que pueden impedir un buen análisis, lo cual juega un papel imprescindible para que los datos utilizados sean de cierta calidad y significado. Así, un buen pre-procesamiento de datos puede simplificar y mejorar el proceso de Minería de Datos.

Los modelos para extraer estos patrones pueden ser predictivos o descriptivos. Los modelos predictivos pretenden, mediante el uso de campos de la base de datos (variables independientes), estimar valores futuros o desconocidos (variables dependientes). Por otro lado, los modelos descriptivos sirven para explorar las propiedades de los datos examinados, no para predecir nuevos datos [4].

Tal y como se muestra en la figura 1, el proceso de minería de datos precisa de otras tareas previas que conforman las siguientes etapas: selección de datos, pre-procesado y transformación.

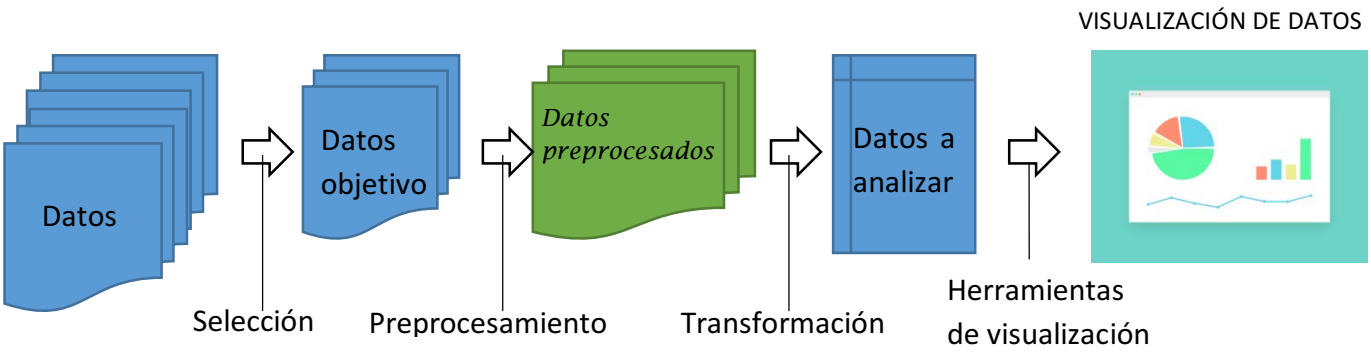



Figura 1- Etapas de un proceso de Minería de Datos

En el presente TFG se ha seguido el anterior proceso, desde la obtención de una gran cantidad de datos hasta la herramienta de visualización que permite analizar dichos datos de forma interactiva.

A lo largo de la presente memoria se explican las tecnologías y procedimientos utilizados para conseguir atravesar cada una de las partes.

## 2.3 Visualización de datos

Actualmente nos encontramos en lo que bien podría denominarse como "Un mundo de datos". Los aspectos de nuestra vida que antes eran efímeros y banales en un contexto productivo ahora se acumulan como datos debido a la digitalización de estos procesos cotidianos. Estos datos se generan a una gran velocidad y en grandes cantidades, y normalmente los procesamos para entender el mundo en que vivimos e incluso tomar decisiones de futuro. Así, en un mundo cada vez más conectado se produce cada vez más información debido al uso de las redes sociales, buscadores o dispositivos electrónicos.

Sin embargo, intentar entender los datos de forma aislada carece de sentido, puesto que éstos no tienen gran significado. De esta manera, la visualización de datos se convierte en una herramienta de interpretación de datos muy potente y necesaria para que los datos adquieran significado.

Generalmente, las personas entendemos o recibimos mejor la información cuando ésta aparece gráficamente. Ahí reside la esencia del porqué es tan importante la visualización de los datos.

Dentro de los **elementos básicos** para la representación de datos se encuentran:

- Gráficas: de barras, líneas, tarta, puntos, etc.
- Mapas: de calor, de agregación, coropletas, etc.
- Tablas: dinámicas, de transiciones, etc.

Por otro lado, existen los **cuadros de mando o *dashboards***, composiciones de visualizaciones individuales que tienen relación entre sí y están relativamente coordinadas. Son muy útiles para el análisis de datos y toma de decisiones [5].

A lo largo de esta memoria, se desarrollan las tecnologías y procesos llevados a cabo para conseguir un cuadro de mando que consigue tratar y transformar datos en bruto a información lista para analizar.

## 2.4 Alternativas

### 2.4.1 Tecnoempleo.com

Tecnoempleo.com es un portal web donde empresas de carácter tecnológico publican sus ofertas de empleo, a las cuales los usuarios pueden inscribirse para ser candidatos dicho puesto empleo.

Actualmente existe en este portal de empleo [6] una herramienta que permite clasificar ofertas de empleo según:

- Top Tecnologías (java, Linux, c#, SQL…)
- Top Funciones Profesionales (Desarrollador web, Consultor, Analista Programador, Redes…)
- Top Países/ Provincias (Alemania, Asturias, Irlanda…)

En dichas clasificaciones se enumeran las tecnologías, funciones profesionales y localizaciones que aparecen con mayor frecuencia entre todas las ofertas existentes en el portal. En el caso de las tecnologías y funciones, aparece el número de ofertas de empleo relativo de la tecnología o función en concreto. También permite el acceso a dichas ofertas de empleo en caso de seleccionar uno de esos criterios (filtros).

En las siguientes figuras se muestra la interfaz de dicha herramienta de filtrado, proporcionando cierta estructura y orden en la clasificación de ofertas, lo cual facilita al usuario una búsqueda exhaustiva:

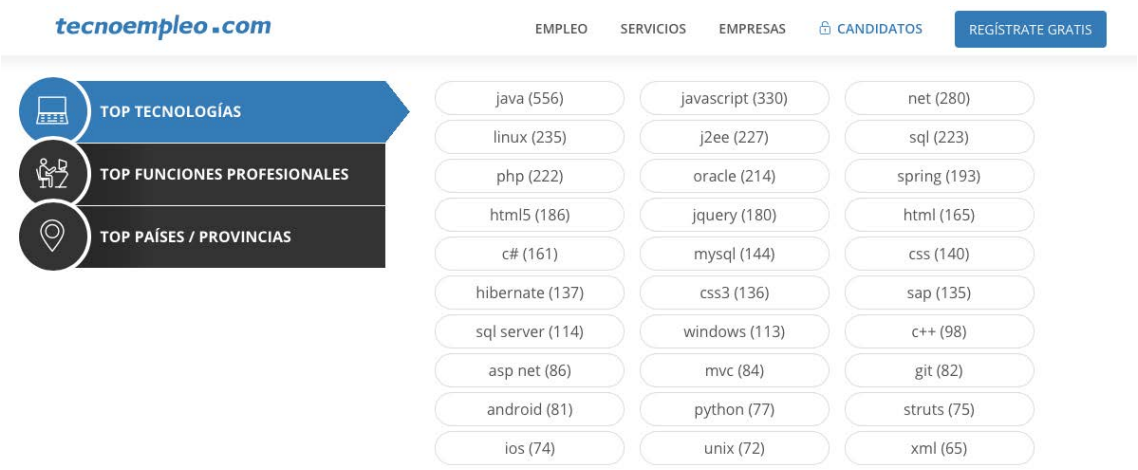



Figura 2- Tops Tecnoempleo

Por otro lado, se puede acceder a informes de Empleo del sector Tecnológico en el cual se permite el acceso a estadísticas de empleo tales como:

- Salario medio según Funciones.
- Experiencia requerida según puestos.

- Empresas que más publican.
- Estadísticas de ofertas en Tecnoempleo.com
- Estadísticas de CVs en Tecnoempleo.com

En dichos informes se expone analíticamente el valor de los filtros indicados, como podemos ver a continuación:

Top Empresas

| Nombre | Empleo | Nombre | Empleo |
|---|---|---|---|
| Krell Consulting & Training | 339 | Ibermatica | 46 |
| HAYS | 225 | Everis Spain | 44 |
| Page Personnel | 198 | Grupo GFI Informática | 43 |
| Altran España | 98 | Indra Software Labs | 43 |
| Michael Page | 98 | Accenture Technology | 41 |
| ALTEN Spain | 72 | Sopra Steria | 41 |

**Figura 3- Top empresas Tecnoempleo**

Top Funciones / Roles - Tecnologías

| Empleo | | Empleo | | | |
|---|---|---|---|---|---|
| Programador | 1.066 | java | 599 | javascript | 417 |
| Analista Programador | 692 | sql | 303 | linux | 294 |
| Desarrollador Web | 364 | net | 278 | oracle | 264 |
| Consultor | 315 | php | 223 | c# | 221 |
| Técnico de Sistemas | 312 | html | 200 | css | 189 |
| Analista | 270 | mysql | 186 | windows | 184 |
| Jefe de Proyecto | 177 | spring | 183 | j2ee | 171 |
| Soporte Técnico | 149 | html5 | 155 | jquery | 151 |
| Administrador | 140 | sap | 149 | git | 149 |
| Helpdesk | 100 | css3 | 133 | angular | 129 |
| Arquitecto TIC | 93 | | | | |

**Figura 4- Top funciones Tecnoempleo**

Salario - Funciones / Roles

| Funciones / Roles | *Salario Bruto/Año | Funciones / Roles | *Salario Bruto/Año |
|---|---|---|---|
| I+D | 31.500 - 41.500 € | Tester | 23.000 - 29.200 € |
| Arquitecto TIC | 29.400 - 39.800 € | Desarrollador Web | 21.700 - 28.500 € |
| Sistemas de Calidad | 27.400 - 35.900 € | Técnico de Gestión | 21.400 - 26.400 € |
| Consultor | 25.800 - 33.300 € | Técnico de Sistemas | 21.400 - 26.200 € |
| Administrador | 26.900 - 33.200 € | Comercial | 21.500 - 24.700 € |
| Formador | 24.900 - 32.800 € | Redes | 20.700 - 24.600 € |
| Analista | 25.500 - 31.900 € | Diseño | 18.700 - 24.000 € |

**Figura 5-Top salarios Tecnoempleo**

Experiencia - Tipo de Contrato

| Experiencia | Empleo | Porcentaje | Tipo de Contrato | Empleo | Porcentaje |
|---|---|---|---|---|---|
| Sin Experiencia | 190 | 6 % | Indefinido | 2.160 | 72 % |
| Menos de un año | 109 | 4 % | Obra o servicio | 146 | 5 % |
| 1 año | 653 | 22 % | Temporal | 158 | 5 % |
| 2 años | 868 | 29 % | Freelance/Autónomo | 15 | 1 % |
| 3 años | 551 | 18 % | Prácticas | 55 | 2 % |
| 3-5 años | 404 | 14 % | A determinar | 449 | 15 % |

Figura 6- Tipo de contrato Tecnoempleo

- **Mejoras**

De las bases de datos (ofertas de empleo), utilizadas para la creación de la herramienta de análisis presentada en este TFG, gran parte de ellas pertenecen a Tecnoempleo.com. Por ello se considera pertinente analizar las posibilidades que este portal web proporciona para que los usuarios reciban información de una manera más interactiva.

En comparación de aquellas con la presente herramienta web, se exponen las siguientes debilidades que remarcan la utilidad de la segunda:

Tabla 1- Comparación con Tecnoempleo

| TECNOEMPLEO.COM | HERRAMIENTA DE VISUALIZACIÓN |
|---|---|
| La herramienta proporcionada por el portal Tecnoempleo.com aparece en la parte última dentro del sitio web (no siendo anunciada en ningún otro lugar), por lo que su uso se ve reducido a aquellos que dedican un tiempo a curiosear en la página. Así, esta herramienta resulta ser poco accesible. | Está centrada en esta herramienta, por lo que su accesibilidad es inmediata y su facilidad de uso evidente. |
| No existe relación ni interactividad entre las diferentes ofertas según su clasificación. | Al interactuar con los diferentes gráficos (clasificación), se producen cambios en los demás gráficos.<br>Herramienta interactiva y dinámica. |
| Los resultados de los diferentes informes o filtros se muestran en forma de tablas. | Los resultados de los diferentes filtros se muestran en forma de gráfico de sectores, mapas, gráfico de barras y otros métodos |

| | |
|---|---|
| | de visualización más atractivos. |
| Solo contiene ofertas de tecnología e informática | Los datos utilizados son ofertas de empleo, no solo del sector de tecnología e informática sino también marketing y otros. |

### 2.4.2 InfoJobs

InfoJobs, de forma semejante a Tecnoempleo, es un portal web para la búsqueda de empleo, pero, en este caso, de un amplio rango de puestos de trabajo o categorías (no sólo de informática y tecnología). Al igual que su competidora, ofrece la posibilidad de realizar filtros en las búsquedas y, cómo herramienta de análisis, ofrece información sobre cuáles son las categorías, provincias o empresas donde el número de ofertas son mayores.

En este proyecto se han utilizado las ofertas de empleo de InfoJobs como fuente de datos para analizar.

En la figura 7, se puede observar el tipo de análisis que la plataforma web ofrece a sus usuarios, siendo simplemente una clasificación según frecuencia de aparición entre categorías, empleas o provincias [7].


Figura 7- Tops InfoJobs

Al igual que Tecnoempleo, InfoJobs no ofrece gran accesibilidad en su web a esta herramienta ni resulta atractiva para los usuarios.

### 2.4.3 Marca empleo

Marca empleo es una página web donde se pueden encontrar ofertas de trabajo, oposiciones, cursos de formación, becas y otras noticias relacionadas existentes en España o también en el extranjero.

Como motor de búsqueda de empleo es muy completa puesto que tiene gran cantidad de información y de gran variedad. Esto es: bolsas de empleo en función de provincias o localidades, para personas con discapacidad, jóvenes, etc.

En esta página aparecen las ofertas distribuidas en bolsas de empleo por lo que no es concluyente su utilidad para realizar un análisis de esta distribución de ofertas.

Lo más parecido a un filtrado y, por tanto, clasificación de las ofertas, es la posibilidad que ofrece de escoger ofertas en función del sector. Por ejemplo: administración/oficinas, sanidad, ingeniería, informática u hostelería. Sin embargo, no ofrece ninguna interfaz o herramienta específica para el análisis de las ofertas de empleo.

Como conclusión, esta herramienta es de carácter informativo, útil para la búsqueda activa de empleo, aunque también poco intuitiva por el exceso de enlaces, pestañas y el modo en que está volcada la información. Por ello, no supone una competencia a la herramienta expuesta en este TFG como herramienta para el análisis de las ofertas de empleo. Aunque si podría contemplarse la opción de ser utilizada para la obtención de datos (como base de datos).

### 2.4.4 LinkedIn

En la actualidad, se entiende *Networking* como la forma de generar contactos o vínculos profesionales con personas, de forma que se crean relaciones entre empresas y profesionales. Así, el 70% de los responsables de selección de las empresas investigan las redes sociales de los candidatos [8]. Una de las redes más utilizadas para llevar a cabo dicha actividad es LinkedIn.

LinkedIn se trata de una red social profesional donde las empresas y profesionales se promocionan y comparten sus intereses e inquietudes profesionales para satisfacer la búsqueda de empleo por parte del profesional o la captación de nuevos empleados por parte de las empresas.

A pesar de que se han detectado un 35% más de contrataciones en esta red en el último año [9], se piensa que LinkedIn se utiliza más para hacer contactos que para encontrar empleo.

Gracias a las herramientas que ofrece, LinkedIn permite a los profesionales observar a la competencia y conocer las exigencias del mercado. Por otro lado, en la

sección de “Empleos” se proporcionan aquellos empleos que pueden ser interesantes para el profesional orientados a su carrera, en función de su perfil e intereses de empleo. Sin embargo, a pesar de que hubiera dotado de robustez a la herramienta desarrollada, estos datos no pudieron ser utilizados como fuente de ofertas de empleo para el proyecto debido a las licencias de uso.

# 3. ANÁLISIS DEL PROBLEMA

## 3.1 Análisis de los requisitos y soluciones

Para la creación de la herramienta de visualización de ofertas de empleo se necesita:

- En primer lugar: un **conjunto de ficheros** que contengan las ofertas de empleo para poder ser procesadas. Se obtendrían resultados más representativos si estas ofertas de empleo provienen de diferentes bases de datos.
- Se ha de concretar qué **criterios** dentro de las ofertas son útiles para ser representados.
- Es necesaria la elección de qué **tipo de gráfico** será utilizado para representar para cada uno de los criterios de las ofertas.
- Se deben tratar los datos y **homogeneizar**los para un correcto comportamiento y representación.
- Se debe marcar una estructura del **sitio web**: qué posición ocupará cada gráfico, qué tamaño o color, entre otros.

# 4- DISEÑO DE LA SOLUCIÓN

En este capítulo se abordará el diseño de la herramienta web en su totalidad, considerando cada una de las fases o partes en las que fue dividida la solución completa para poder satisfacer los requisitos expuestos en el anterior capítulo.

## 4.1. Esquema de la solución

El proceso de creación de la herramienta de visualización de ofertas de empleo se divide en tres fases diferenciadas:

- **Descarga de datos:** Recolección periódica de ficheros que contienen las ofertas de empleo. En el caso de InfoJobs se utiliza su API para la descarga de datos, mientras que en el caso de Tecnoempleo se descarga un archivo de formato XML que contiene las ofertas de empleo publicadas cada día.
- **Pre-procesado de datos:** Tratamiento de los datos recolectados para que puedan ser posteriormente utilizados en su análisis y visualización.
- **Visualización:** Uso de tecnologías existentes para la representación gráfica de los datos previamente tratados.

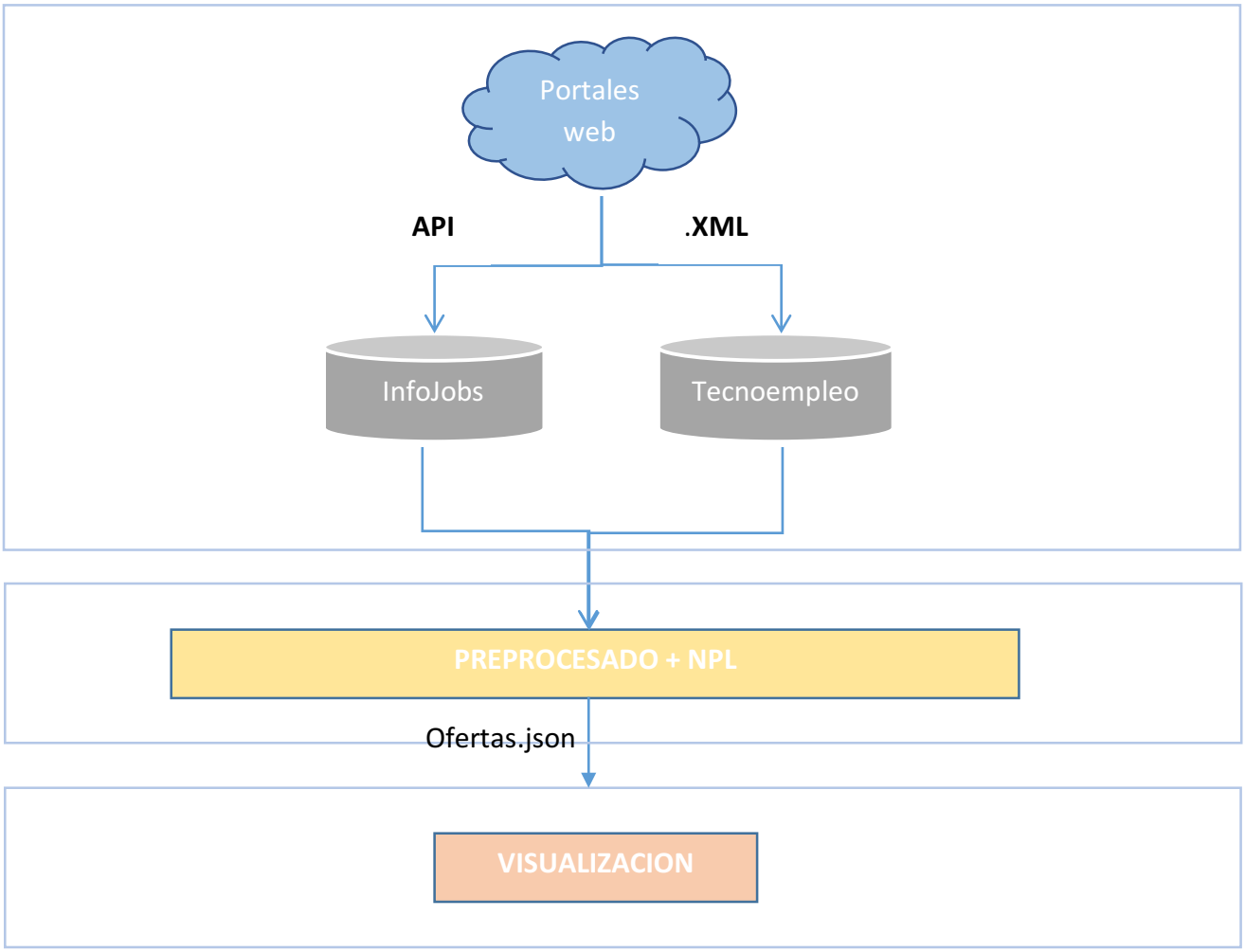



Figura 8- Esquema de las fases de la solución

## 4.2 Análisis de las herramientas

En este apartado se van a analizar las distintas tecnologías utilizadas durante el desarrollo del proyecto, estudiando su funcionalidad y explicando su elección frente a otras opciones. Se han organizado todas las tecnologías empleadas en tres bloques funcionales. Estos bloques son los siguientes: en primer lugar, Tecnologías utilizadas para la obtención de datos; el segundo bloque, Tecnologías para el procesamiento de los datos; y en último lugar, Tecnologías para la visualización de datos.

### 4.2.1 Tecnologías para la obtención de datos

Para la descarga y recolección de las ofertas de empleo se utilizan scripts escritos en el lenguaje de programación Python. Se utiliza este lenguaje ya que hace muy sencilla la descarga y almacenamiento de los contenidos de una página web mediante el procesamiento automático de ellas, para lo cual tiene distintos estándares.

Durante este bloque se han utilizado muchos de los módulos de Python que se utilizaron también en el bloque de procesamiento de datos, por lo que serán expuestos en el siguiente punto.

### 4.2.2 Tecnologías para el procesamiento de datos

#### *4.2.2.1 JSON*

JSON [10], acrónimo de JavaScript Object Notation, es un formato ligero de intercambio de datos. Se utiliza para representar datos estructurados que siguen la sintaxis de objeto de JavaScript y para transmitir datos en aplicaciones web. Está constituido por dos estructuras:

- Una colección de pares nombre/valor, conocido como objeto, registro, estructura, tabla hash, diccionario o lista en otros lenguajes.
- Una lista ordenada de valores, implementada en otros lenguajes como vectores, secuencias o listas.

A continuación, se muestra un ejemplo de mensaje con formato JSON:

```
{ "menuitem": [
      {"value": "New", "onclick": "CreateNewDoc()"},
      {"value": "Open", "onclick": "OpenDoc()"},
      {"value": "Close", "onclick": "CloseDoc()"}
    ]
  }
```

De este mensaje se puede extraer la colección de pares nombre-valor:

Tabla 2- Pares nombre/valor de un ejemplo JSON

| Nombre | Valor-Nombre | Valor |
|---|---|---|
| **Menuitem[0]** | value | New |
| | onclick | CreateNewDoc() |
| **Menuitem[1]** | value | Open |
| | onclick | OpenDoc() |
| **Menuitem[2]** | value | Close |
| | onclick | CloseDoc() |

Comparación entre JSON y XML

En este trabajo, ha sido utilizado el formato JSON y no **XML**, ya que sólo era necesario almacenar datos de tipo texto y números (tipo int o entero), de forma que no era pertinente la ventaja de extensibilidad que ofrece XML (lo cual lo hace más difícil de leer). Aunque el formato JSON sea más restrictivo, es más legible, lo cual es ventajoso al tratar con una gran cantidad de datos. Por ello, es más fácil importar datos desde un fichero JSON a JavaScript, Python u otros lenguajes que en el caso de ficheros XML.

Como conclusión, por un lado, XML [11] ofrece una mayor extensibilidad de tipo de objetos que se pueden transmitir a cambio de una menor legibilidad y mayor dificultad de importación desde otros lenguajes. Como, en este caso, los datos transferidos son datos sin formato y tradicionales (texto plano), fue escogido el formato JSON para la transferencia de los datos, el conjunto de las ofertas de empleo, y ser utilizados como entrada para la parte del proyecto que implementa la representación gráfica.

#### *4.2.1.2 Python*

Después de descargar los datos, se necesitaba una herramienta robusta y fiable que permitiese procesar, filtrar y manipular toda esta información. Para ello se ha utilizado el lenguaje de programación Python.

Python [12] es un lenguaje de programación que soporta programación orientada a objetos, programación funcional o imperativa. Con lo cual, es un lenguaje multiparadigma y multiplataforma, además de un lenguaje interpretado y

que usa tipado dinámico (una misma variable puede tomar distintos tipos de valores según el momento). Entre las ventajas que suponen programar en Python se encuentran:

- Es libre y ofrece código abierto: la filosofía Python es que el código es legible y transparente. Sigue principios como: ser práctico, plano, explícito, simple o que la implementación debe ser fácil de explicar.
- Calidad en su sintaxis: fuerza al desarrollador a crear un código legible mediante, por ejemplo, la indentación o sangrado del código.
- Ofrece una gran cantidad de librerías disponibles para el tratamiento de ficheros y grandes cantidades de datos.

Se ha trabajado sobre *Jupyter Notebook*, un entorno de desarrollo (IDE) que funciona sobre la plataforma de distribución *Anaconda-Navigator*, el cual posee una colección de más de 720 paquetes de código abierto y más de 1.000 paquetes con librerías útiles para la ciencia de datos [13]. Una de las ventajas que presenta Jupyter Notebook y que hizo que fuera elegida frente a otras opciones es que permite la ejecución del código en el propio editor.

Comparación entre Python y R

Previa implementación del presente proyecto se afrontó la elección de qué lenguaje de programación sería más adecuado para utilizar en el procesamiento de los datos. Por ello, se tuvo en cuenta la posibilidad de utilizar el lenguaje de programación R.

R [14] es un lenguaje de programación de código abierto cuyo objetivo es conseguir un análisis de los datos, estadísticas y modelos gráficos mejores y más fáciles de utilizar.

Tanto Python como R son muy útiles para el análisis de datos, sin embargo, Python fue el lenguaje elegido debido a que su curva de aprendizaje es relativamente baja y gradual, gracias a su legibilidad y simplicidad. Además, es preferible el uso de Python cuando las tareas de análisis de datos deben integrarse con otras aplicaciones o servicios. Por otro lado, existen numerosas librerías de Python que facilitan el procesamiento de datos y de las cuales se puede obtener su documentación detallada y ejemplos de uso en la web.

A continuación, se puede observar la popularidad de ambos lenguajes entre 2013 y 2015 [15]. En la figura 9 se muestra como Python (en azul) es más utilizado, pues es un lenguaje de propósito general, mientras el uso de R (en naranja) queda limitado al entorno de *Data Science*, por lo cual se consideró más útil el aprendizaje y uso de Python frente a R.

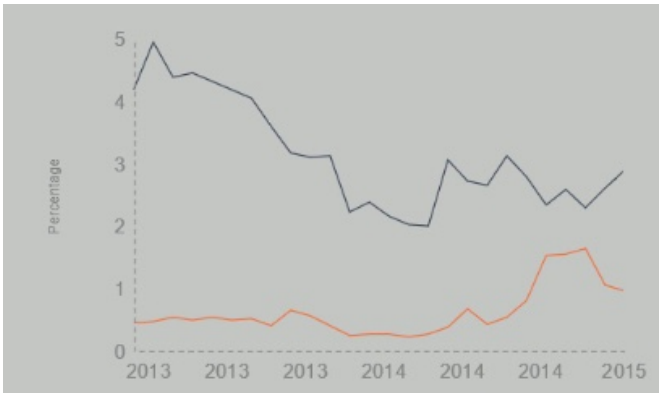



Figura 9- Popularidad de R y Python entre 2013 y 2015 (Índice Tiobe).

Módulos de Python utilizados:

A continuación, van a ser presentados los módulos de Python utilizados para el procesamiento de datos durante el desarrollo del proyecto.

### *Modulo OS*

Permite acceder a funcionalidades dependientes del sistema operativo con el que se trabaje. En este caso se han utilizado aquellas funcionalidades que permiten manipular la estructura de directorios (para leer y escribir archivos) como, por ejemplo, *os.listdir* u *os.path*.

### *Módulo lxml*

Implementa una API simple y eficiente para analizar y crear datos XML [16]. Se utilizó en este proyecto para importar los datos leyendo de un archivo en formato *xml*. Estos archivos son las ofertas de empleo provenientes de Tecnoempleo.

### *Módulo JSON*

JSON (*JavaScript Object Notation*) es una de las librerías para el manejo de datos de Internet. Esta librería se utiliza como codificador y decodificador de JSON [17]. En este proyecto se utilizó esta librería para deserializar (*json.loads*) los archivos de texto que contenían las ofertas de empleo en formato JSON provenientes de InfoJobs. Una vez fueron procesadas dichas ofertas, junto con las ofertas provenientes de Tecnoempleo, se serializaron todos los datos procesados en un archivo de formato JSON mediante el uso de *json.dumps*().

### *Módulo NLTK*

Se utiliza para trabajar con cadenas de texto o datos en lenguaje humano natural [18]. Proporciona interfaces fáciles de usar y un conjunto de bibliotecas de procesamiento de texto natural para la clasificación, derivación, análisis,

tokenización, etc. Es un proyecto gratuito, de código abierto e impulsado por la comunidad.

*Módulo scikit-learn*

Proporciona herramientas simples y eficientes para en análisis de datos y la minería de datos (machine learning). Está construido sobre NumPy, SciPy y matplotlib. Además, es de código abierto y utilizable comercialmente. Entre sus usos se encuentran: clasificación, regresión, agrupación o pre-procesamiento de objetos [19].

### 4.2.3 Tecnologías para la visualización de datos

Durante todo el proceso de desarrollo de la fase de visualización de datos ha sido utilizado un editor de textos llamado *TextWrangler.*

#### *4.2.3.1 HTML (HyperText Markup Language)*

HTML, formalmente HTML5, es un lenguaje de marcado para la elaboración de páginas web. Define una estructura básica y un código para la definición del contenido de una página web como puede ser: texto, imágenes, scripts (como JavaScript), entre otros. De esta manera, HTML se utiliza en el proyecto para organizar el contenido que muestran los navegadores web.

El **DOM** (Modelo de Objetos del Documento) hace referencia a la estructura jerárquica de HTML donde cada una de las etiquetas corresponde a un elemento y los elementos están relacionados entre sí, lo cual permite referenciar y manipular dichos elementos. Los navegadores web descomponen esta estructura para interpretar el contenido de la página.

#### *4.2.3.2 CSS*

Es un lenguaje de estilo que define la presentación de los documentos escritos en HTML, lo relativo a fuentes, colores, márgenes, alturas y anchuras, etc. La separación de la estructura del documento y su representación facilita la modificación de los estilos. Así CSS proporciona un control más preciso de la presentación.

CSS utiliza selectores para marcar el estilo de cada uno de los elementos que forman parte de la estructura del código HTML. De esta forma, los selectores identifican elementos específicos a los cuales se aplicarán los estilos.

#### *4.2.3.3 JavaScript*

Es un lenguaje de programación interpretado, se define como orientado a objetos y guiado por eventos. JavaScript permite crear nuevo contenido dinámico, controlar archivos multimedia, crear imágenes animadas y muchas otras cosas.

El lenguaje JavaScript [20] es ejecutado por el motor del navegador de JavaScript después de que el código HTML y CSS se hayan mezclado dentro de la página web, estableciendo un estilo y estructura en ella, sobre lo cual funciona JavaScript, dotando de interactividad a la página web.

JavaScript permite crear páginas web dinámicas mediante la manipulación del DOM después de que la página ya haya sido cargada en el navegador. Son estas propiedades dinámicas las que han hecho que sea utilizado este lenguaje para la programación de la herramienta de análisis de ofertas de empleo.

#### *4.2.3.4 SVG (Scalable vector graphics)*

Los Gráficos Vectoriales Redimensionables o SVG es un formato de imagen basado en texto. Proporciona una serie de facilidades que hacen que generar y manipular imágenes sea más consistente y rápido que haciéndolo con HTML.

Cada imagen SVG [21] se define con un código de marcado similar a HTML que puede incluirse directamente en cualquier documento HTML o insertarse dinámicamente en el DOM.

SVG es uno de los estándares más utilizados para crear gráficos en 2D [22].

#### *4.2.3.5 D3: Data-Driven Documents.*

D3.js [23] es una librería de JavaScript muy utilizada y reconocida para la visualización de datos que utiliza otras tecnologías como HTML, SVG y CSS. D3 permite la manipulación e inspección directa del DOM. Además, con D3, el diseñador puede asociar selectivamente datos de entrada con elementos del DOM, aplicando transformaciones dinámicas para generar y/o modificar su contenido. De esta manera, gracias a las transformaciones de D3 se dota de animación e interacción a las representaciones de los datos [24].

D3 puede leer JSON y GeoJSON con facilidad, por ello fueron elegidos dichos formatos para los datos de entrada en el proyecto.

Por otro lado, D3 da solución a la manipulación eficiente de documentos basados en datos, el cual suponía un problema en el contexto de la visualización de datos. Por esta razón D3 es considerado un *kernel* en visualización más que un simple *framework*.

Los principales elementos que presenta d3 son:

- *Selecciones*: se refiere al conjunto filtrado de datos que han sido asociados a elementos del documento (DOM).
  En la siguiente figura se muestra un ejemplo de un elemento de tipo “p” (párrafo) y sus datos asociados.

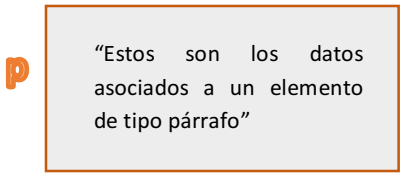



Figura 10- Ejemplo de Selección

- *Operadores*: modifican el contenido de los elementos seleccionados de forma instantánea.
- Enlaces de datos: enlazan datos de entrada a elementos, habilitando aquellos operadores que dependen de dichos datos. También producen sub-selecciones para la creación y destrucción de elementos asociados a datos (entrada y salida) o actualización de los elementos con propiedades dinámicas. Por defecto, los datos son asociados a elementos o nodos por orden.
  Cuando llegan nuevos datos (en azul) y se asocian a elementos antiguos (en naranja) resultan tres sub-selecciones: entrada, actualización y salida. Este es un proceso que ocurre en el caso de realizar filtros en selecciones.

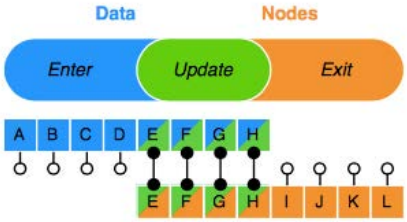



Figura 11- enlace de datos en d3.js

Fuente: http://vis.stanford.edu/files/2011-D3-InfoVis.pdf

- Transiciones: modifican el contenido de las selecciones de forma animada, intercalando atributos y estilos de forma asíncrona a lo largo del tiempo.
- Manejadores de eventos: son un tipo especial de operadores que actúan en respuesta a acciones del usuario y permiten la interacción.

El proceso completo, en el que se encuentran implicados los anteriores elementos, es el siguiente: se cargan los datos en la memoria del navegador; se asignan esos datos a elementos dentro de la página y se van añadiendo nuevos elementos si es necesario; se transforman dichos elementos interpretando cada dato de cada elemento y estableciendo sus correspondientes propiedades gráficas; y se crean transiciones entre elementos en diferentes estados en función de las peticiones del usuario (interactividad). Este último paso es el más importante, puesto que d3 permite escoger qué acciones del usuario provocarán un cambio en la representación de los datos y qué cambio será.

Uno de los objetivos de este proyecto planteados al comienzo de la memoria era el de crear una herramienta interactiva que, junto con la gran cantidad de datos obtenidos, resulta en la necesidad de usar herramientas que faciliten alcanzar dicho objetivo. Por ello, se han utilizado otras herramientas que están construidas sobre D3:

- Crossfilter :

  Es una librería de JavaScript escrita fundamentalmente por Mike Bostok.
  Esta librería es gratuita y permite trabajar con un complejo y gran conjunto de datos rápidamente, de forma que puedan ser filtrados fácilmente o se puedan calcular sumas dentro de este conjunto.

  La principal función de Crossfilter (*Filtros cruzados)* [25] es agrupar los datos en conjuntos y calcular la suma de estos conjuntos. Así, puesto que en nuestro conjunto de datos cada objeto (oferta de empleo) tiene unos campos, se puede encontrar rápidamente la cantidad total de objetos que poseen un valor concreto para uno de esos campos. La forma de acceder a cada uno de esos campos en Crossfilter es a través de una **dimensión.**
  Por ejemplo, para acceder al campo "provincia" de una oferta de empleo creamos la siguiente dimensión:

```
var ndx=crossfilter(ofertasJson) //se define el filtro cruzado
cityDimension=ndx.dimension(function(d){
       return d.province;})
```

A pesar de tener esta clasificación hecha, hay que decirle a Crossfilter cómo queremos que agrupe dichos ítems, esto se hace mediante el **grupo**. En este caso, queremos una suma de todas aquellas ofertas de trabajo que sean de la misma provincia, por lo que buscamos simplemente la suma de la cantidad de cada elemento:

```
cityGroup= cityDimension.group()
```

La **ventaja** de Crossfilter es que al aplicar un filtro se excluyen elementos de nuestro conjunto de datos, con lo que la dimensión y el grupo varían, siendo ésta la clave del dinamismo de la visualización de nuestros datos. Esta ventaja se consigue gracias a las funciones *add(elemento)* y *reduce()*, que agregan nuevos elementos o eliminan aquellos que coinciden con el filtro cruzado, respectivamente.

- DC.js:

  Es la abreviatura de “gráficos dimensionales”, es una librería optimizada para los grandes conjuntos de datos [26].
  El objetivo de esta librería es mostrar fácil y rápidamente los resultados de los filtros cruzados (Crossfilter) mediante diferentes tipos de **gráficos**. La mayor ventaja que ofrece esta librería es que los gráficos son dinámicos, de forma que, al hacer clic en alguno de los gráficos, éste filtra ese conjunto de datos con el grupo seleccionado y actualiza el gráfico.

  Esta librería se apoya en Crossfilter, ya que necesita un conjunto de datos de Crossfilter, una dimensión y un grupo sobre los cuales se puedan crear los gráficos dinámicos. También utiliza d3.js para representar los gráficos en formato CSS compatible con SVG. [24]

  **Comparación con otras herramientas:**

  En el presente proyecto se ha decidido utilizar las anteriores herramientas basadas en D3 gracias a que es muy útil para aprender cómo las herramientas de visualización de datos funcionan en su interior. Actualmente existen muchas plataformas que permiten elaborar gráficos de forma rápida y sencilla a partir de un conjunto de datos, sin embargo, no sería de gran utilidad obtener dichos gráficos sin saber cómo son tratados y manejados los datos antes de llegar al resultado final. Algunas de estas plataformas son:

  - QlikSense

    Es una potente plataforma que permite al usuario analizar los datos, independientemente de su tipo y fuente, creando visualizaciones que facilitan la interpretación del dato y la toma de decisiones.
    La principal desventaja de esta herramienta es que es de pago, además esta herramienta puede ser utilizada sin necesidad de poseer ninguna habilidad profesional, por lo cual no resulta interesante.
  - Google Charts

    Es una biblioteca propiedad de Google [27] que permite generar gráficos automáticamente para después insertarlos en una página web. Proporciona distintos tipos de gráficos como: gráficos de barras, gráficos de sectores, así como la posibilidad de darles etiquetas y colores.
    Es una buena herramienta para crear gráficos de forma sencilla, sin embargo, tiene algunas restricciones como: límite en los píxeles de altura y anchura de los gráficos o límite en el número de consultas (5.000) por usuario en 24 horas. Además, ofrece algunos componentes para la creación de *dashboards* interactivos, pero de forma limitada.

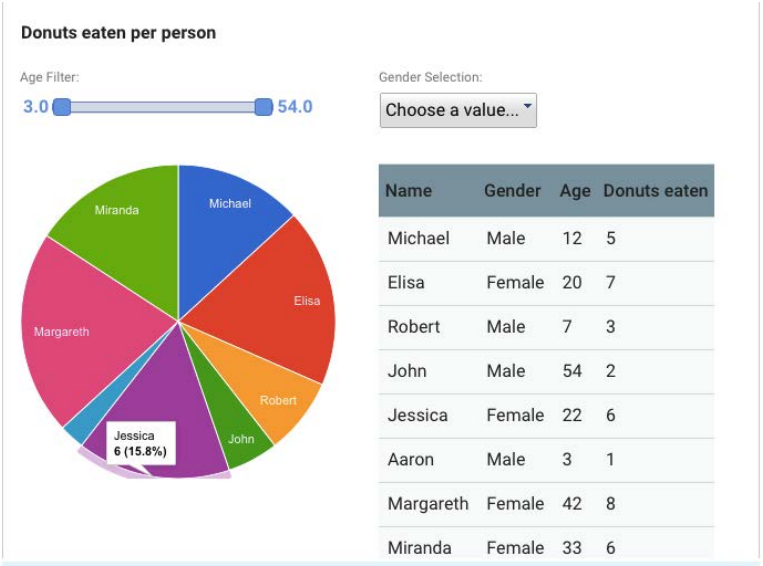



Figura 12- Gráficos interactivos creados con Google Charts.

Fuente: https://developers.google.com/chart/

## 4.3 Desarrollo de la solución

Una vez se conocen las herramientas utilizadas durante el proyecto, van a ser explicadas con mayor profundidad cada una de las fases que lo conforman:

### 4.3.1 Descarga de datos

Esta primera parte es la más automática y sencilla. Consiste en la descarga de datos ubicados en las bases de datos de los portales de empleo web InfoJobs y Tecnoempleo.

Por un lado, InfoJobs tiene a disposición de cualquier usuario una API que puede ser utilizada para crear aplicaciones web, de móvil o de escritorio, entre otras cosas. Esta API es la que fue utilizada, junto con un script de Python, para la descarga de los datos de dicho portal.

Las ofertas de empleo de InfoJobs son descargadas diariamente, de forma que la fecha de subida mínima de las ofertas sea, como mucho, un día anterior al de la consulta actual. Y la fecha de subida máxima de las ofertas sea, como mucho, el día actual. Estos dos límites temporales se utilizan como parte de la URL en la petición al servidor de forma que devuelva la página web que contenga solo las ofertas que cumplan dichos límites.

Python permite almacenar el contenido de la página web en formato JSON. Además, para cada día, las ofertas son clasificadas en función de su categoría: Informática y Telecomunicaciones (IFC), Marketing y comunicación (MC) o Diseño y Artes Gráficas (ARG).

En la siguiente figura se detalla la jerarquía de directorios en la que se almacenan las ofertas en formato TXT.

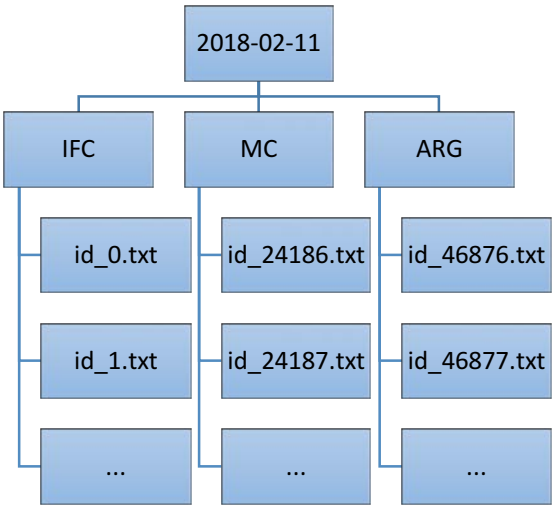



Figura 13- Jerarquía en el directorio de ofertas de InfoJobs

Por otro lado, los datos del portal Tecnoempleo son almacenados en formato XML desde el propio sitio web. Gracias a ello, el script de Python para el almacenamiento de los datos contenidos en la página web es bastante más sencillo, puesto que basta con leer su contenido y guardarlo en un fichero que contendrá todas las ofertas diarias sin ningún tipo de clasificación por categoría.

Al igual que en el anterior caso, las ofertas de Tecnoempleo se consultan y almacenan diariamente.

### 4.3.2 Pre-procesado de datos y procesamiento de lenguaje natural

Tal y como un reciente artículo menciona, un 90% de los datos de toda la historia se han generado en los últimos cinco años [28]. Y esto no ha hecho más que empezar. En la era de la transformación digital el mayor reto es transformar esa gran cantidad de datos en información con valor, es decir, una cantidad masiva de datos no sirve de absolutamente nada si ésta no se **procesa**.

Los datos de los que se parte no son más que información (ordenada pero desorganizada) que deben ser manipulados para que sean útiles y, tras el proceso de visualización, en un formato visible y comprensible.

En esta fase del proyecto se procesan los datos de entrada para obtener, como datos de salida, el archivo en formato JSON que recibirá la herramienta de visualización.

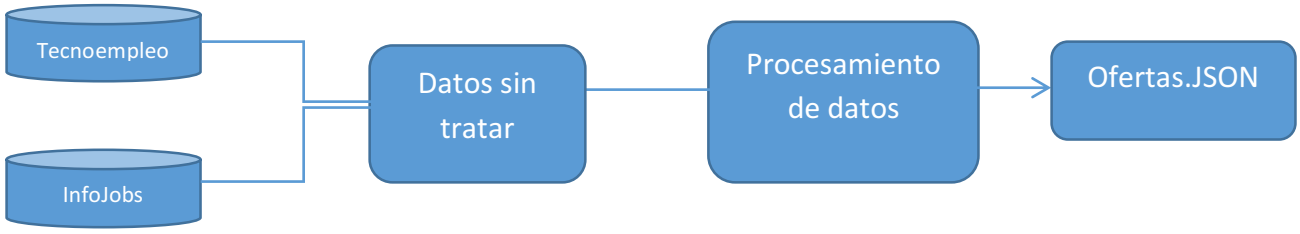



Figura 14- Esquema de procesos en el tratamiento de datos

La fase de pre-procesamiento de datos incluye las siguientes tareas:

- Recolección e integración de datos (véase 4.3.1-DESCARGA DE DATOS)
- Limpieza de los datos
- Transformación de los datos
- Reducción de los datos

### a) Limpieza de los datos

Durante esta fase se eliminaron todos aquellos datos que no iban a ser utilizados en posteriores fases del proyecto.

En primer lugar, posterior a la recolección de datos, se almacenaron en memoria archivos temporales o de configuración que nada aportaban sobre el contenido de los datos. Así, hubo que prescindir de ellos en la selección de los datos útiles mediante una navegación adecuada por los directorios que almacenaban los datos.

Los datos recolectados directamente de las fuentes utilizadas no tenían el mismo formato.


```
{"title":"Q&A Tester","id":"94b3cacb4841a98efba4005b7998ca","state":
1,"creationDate":"2018-02-04T14:46:41.000+0000","updateDate":"2018-02-11T15:00:00.000+0000","city":"Barcelona","externalUrlForm":"","bloc
ked":false,"applications":43,"province":{"id":9,"value":"Barcelona"},"experienceMin":{"id":6,"value":"Al menos 2 años"},"category":{"id":
150,"value":"Informática y telecomunicaciones"},"subcategories":[{"id":3108,"value":"Calidad","order":20,"key":"calidad","parent":
150}],"studiesMin":{"id":30,"value":"Formación Profesional Grado Medio"},"residence":{"id":0,"value":"(Seleccionar)"},"country":{"id":
17,"value":"España"},"contractType":{"id":1,"value":"Indefinido"},"journey":{"id":1,"value":"Completa"},"profile":
{"id":"961656524526202150523215750203","name":"Softmachine","description":"Lider soluciones de Control Horario y Accesos de Alto
Rendimiento.Mercado nacional e internacional.","province":{"id":9,"value":"Barcelona"},"web":"http://www.softmachine.es","numberWorkers":
25,"url":"http://www.softmachine.es","corporateWebsiteUrl":"/softmachine/em-
i961656524526202150523215750203","websiteUrl":"","hidden":false},"cityPD":7824,"zipCode":"08013","latitude":0.0,"longitude":
0.0,"exactLocation":false,"department":"","vacancies":1,"minRequirements":"Herramientas para la ejecución y seguimiento de Testing.
\r\nConocimientos de BBDD SQL. \r\nConocimiento de diferentes Tipos de Testing \r\nBackground de Desarrollador y/o conocimientos de
lógica.\r\nIdioma inglés a nivel de escritura, lectura y conversación.","description":"Técnico de Q&A y Testing funcional de
aplicaciones\r\n\r\nPrincipales Responsabilidades: \r\n\r\nEjecución de tareas del Plan de Testing \r\nGeneración de casos de prueba.
\r\nGeneración documentación que respaldan las pruebas realizadas.\r\nGarantizar el correcto funcionamiento de la aplicación a nivel
funcional. \r\nDocumentación de la aplicación.\r\n\r\nCompetencias: \r\n\r\nGestión por proyectos, organización y planificación de
tareas.\r\nComunicación verbal y escrita.\r\nAnálisis funcional.\r\nGeneración y ejecución de casos de Testing (Test Cases).\r\nTrabajo
en equipo.\r\n\r\n\r\nConocimientos / Experiencia Requerida\r\n:\r\nExperiencia en ejecución de Testing.\r\nHerramientas para la
ejecución y seguimiento de Testing.\r\nBBDD SQL. \r\nDiferentes Tipos de Testing \r\nBackground de Desarrollador y/o conocimientos de
lógica.\r\n\r\nConocimientos / Experiencia Deseable:\r\n \r\nMetodologías Ágiles.\r\nHerramientas de gestión y automatización de pruebas.
\r\nHerramientas de pruebas de estrés y carga.\t\r\nRepositorio de código y control de versiones con SVN\r\n\r\nFormación:
\r\n\r\nFormación profesional en informática o Ingeniería Técnica o Superior en Informática. \r\nIdioma inglés a nivel de escritura,
lectura y conversación.\r\n\r\nSe ofrece: \r\n\r\nIncorporación inmediata en el área de Q&A, integradas en el departamento de I+D en el
centro de trabajo de Barcelona.\r\nPlan de formación interno sobre reglas de negocio.\r\nTrabajo en equipo.\r\nDesarrollo profesional a
largo plazo.\r\nJornada laboral flexible.\r\nPosibilidad de tele trabajo en coordinación con el resto del área.\r\nBuen ambiente de
trabajo dinámico y colaborativo.","desiredRequirements":"","commissions":"Posibilidad de tele
trabajo","contractDuration":"","referenceId":"","detailedStudiesId":-2,"studying":false,"showPay":false,"schedule":"Flexible","jobLevel":
{"id":2,"value":"Empleado/a"},"staffInCharge":{"id":1,"value":"0"},"hasKillerQuestions":1,"hasOpenQuestions":1,"upsellings":
{"highlightColor":false,"highlightUrgent":false,"highlightHomeMonth":false,"highlightHomeWeek":false,"highlightSubcategory":true,"highlig
htLogo":false,"highlightStandingOffer":false},"link":"https://www.infojobs.net/barcelona/q-tester/of-
i94b3cacb4841a98efba4005b7998ca","active":true,"archived":false,"deleted":false,"disponibleForFullVisualization":true,"availableForVisual
ization":true}
```


Figura 15- Ejemplo del texto en bruto de una oferta de InfoJobs

```
<?xml version="1.0" encoding="utf-8" ?>
<ofertas>
<ad>
<id><![CDATA[3afeeace3q1c94bb809a]]></id>
<title><![CDATA[Administrador de sistemas informáticos, Málaga]]></title>
<url><![CDATA[https://www.tecnoempleo.com/administrador-de-sistemas-informaticos-malaga/wintel-unix-linux-vmware-hyp/
rf-3afeeace3q1c94bb809a?lang=es&utm_source=general]]></url>
<content><![CDATA[Administrador de sistemas informáticos en Málaga
Se precisan con urgencia administradores de sistemas informáticos altamente cualificados para nuestros proyectos en toda
Andalucía.

IMPRESCINDIBLES :
- Titulación universitaria en ámbito informático y Telecomunicaciones
- Experiencia de al menos 4 años demostrables en el sector TIC

OFRECEMOS:
- Contrato Indefinido.
- Remuneración Competitiva a Negociar.

PERFIL REQUERIDO - GESTIÓN Y ADMINISTRACIÓN DE :
- Plataformas Wintel y UNIX/LINUX.
- entornos de virtualización (VMware y Hyperv).
- Electrónica de red LAN/WAN, comunicaciones y seguridad perimetral  (HP, CISCO).
- Controladores de dominio (DNS, WINS, DHCP).
- BD y almacenamiento (Oracle, SQL server y NetBackup).

VALORABLES  :
- Cisco CCNA.
- Administración de Windows Server 2012.
- Administración de Bases de datos Oracle 11g.
- ITIL Foundation.


wintel, unix, linux, , vmware, hyperv, , lan, wan, HP, Cisco, Dns, Wins, Dhcp,oracle, Oracle, SQLServer, NetBackup
]]></content>
<region><![CDATA[Málaga]]></region>
<city><![CDATA[Málaga]]></city>
<country><![CDATA[España]]></country>
<salary><![CDATA[Entre 16000 Euros y 24000 Euros Bruto/año]]></salary>
<salary_numeric><![CDATA[20000]]></salary_numeric>
<experience><![CDATA[3-5 anos]]></experience>
<contract><![CDATA[Indefinido]]></contract>
<requirements><![CDATA[wintel, unix, linux, , vmware, hyperv, , lan, wan, HP, Cisco, Dns, Wins, Dhcp,oracle, Oracle, SQLServer,
NetBackup]]></requirements>
<company><![CDATA[Pulsia Technologies]]></company>
<category><![CDATA[Informatica y Telecomunicaciones]]></category>
<date><![CDATA[06/02/2018]]></date>
<time><![CDATA[20:01]]></time>
</ad>
```

Figura 16- Ejemplo del texto en bruto de una oferta de Tecnoempleo

Como se puede observar en las anteriores imágenes, las ofertas de Infojobs tenían un formato estilo JSON, mientras que las ofertas de Tecnoempleo tenían un formato XML.

### b) Transformación de datos

Una de las primeras tareas llevadas a cabo fue normalizar los datos de forma que todos los atributos tuvieran el mismo formato, independientemente de su origen. Gracias a ello, ante una oferta particular como entrada del bloque, se obtiene un diccionario (formato JSON) homogéneo para todas las ofertas, con el mismo tipo, formato y cantidad de atributos, por ejemplo: de fechas, salario o códigos de provincias.

Para esta homogeneización fue necesario realizar la tarea de **relleno de valores faltantes.** Los valores faltantes en un conjunto de datos pueden presentar inconsistencias en el posterior análisis de los datos. Así, estos valores faltantes,

conocidos como *missing values*, se presentan por factores como la recopilación (en este caso, uso de distintas bases de datos) y son reemplazados por valores normalizados. Por ejemplo: Los datos provenientes de la base de datos de InfoJobs carecen de campo "salario", el cual se consideró relevante de analizar. Por ello, se rellenaron estos datos con el valor normalizado "Sin especificar", también existente en datos de Tecnoempleo.

Otra tarea desempeñada dentro de la transformación de datos fue la **discretización de datos**, que consiste en transformar atributos numéricos y representarlos como un intervalo. Esto se utilizó para los campos de: "Experiencia mínima requerida", "Salario" o "Fechas de creación".

Por ejemplo: el campo de "experiencia mínima requerida" se procesó de forma que si su valor era (3-5), se le atribuía un valor de "Al menos 3 años".

### c) Selección de atributos

Para elegir la estructura de datos que sirviera de representación de una oferta de empleo se consideró, del total de atributos que éstas incluían, aquellos que fueran más relevantes para la posterior visualización. Es decir, aquellos datos de los cuales se pudiera obtener información lo más representativa posible. Además, se tuvo en cuenta que la mayoría los atributos elegidos como representativos aparecieran en las ofertas de ambas bases de datos.

Así, el archivo JSON que resulta como entrada para la herramienta de visualización es una **lista de diccionarios**, donde cada diccionario representa una oferta de empleo, con las siguientes claves y posibles valores:

Tabla 3- Atributos de una oferta y sus posibles valores

| CLAVE | POSIBLES VALORES |
|---|---|
| **Fecha de creación** | DD/MM/YYYY |
| **Ciudad** | Nombre de la ciudad en la que se oferta el puesto de trabajo |
| **Provincia** | Número (identificador) de la provincia en la que se oferta el puesto de trabajo |
| **Experiencia** | Al menos n año(s)<br><br>No Requerida |
| **Categoría** | Informática y Telecomunicaciones<br><br>Marketing y Comunicación |

| | |
|---|---|
| | Diseño y Artes Gráficas |
| **País** | Nombre del país en el que se oferta el puesto de trabajo |
| **Tipo de contrato** | Otros contratos<br><br>De duración determinada<br><br>Indefinido |
| **Jornada** | Completa<br><br>Indiferente<br><br>Parcial-Mañana, Parcial-Tarde, Parcial-Noche, Parcial-Indiferente<br><br>Intensiva, Intensiva-Indiferente<br><br>'nd' (Tecnoempleo) |
| **Número de vacantes** | Número entero del número de vacantes ofertadas. Por ejemplo: 1,2,4 o 6. |
| **Requerimientos mínimos** | Se exponen todos los requisitos mínimos, por ejemplo: lenguajes de programación, idiomas, dominio de programas, disponibilidad, valores personales, etc. |
| **Descripción de la oferta** | Contiene información como: funciones que se desarrollarán en el puesto, horario, salario, requerimientos, ciudad, etc. |
| **Salario** | Sin especificar<br><br>Menos de 10.000<br><br>20.000-30.000<br><br>30.000-40.000<br><br>Más de 40.000 |
| **Fuente de datos** | Infojobs<br><br>Tecnoempleo |

En la próxima tabla se tiene un ejemplo de una oferta:

Tabla 4- Ejemplo de una oferta en estructura de diccionario

| CLAVE | VALOR |
|---|---|
| **Fecha de creación** | 05/02/2018 |
| **Ciudad** | Valencia |
| **Provincia** | 49 |
| **Experiencia mínima** | Al menos 2 años |
| **Categoría** | Diseño y artes gráficas |
| **País** | España |
| **Tipo de contrato** | Indefinido |
| **Jornada** | Completa |
| **Número de vacantes** | 1 |
| **Requerimientos mínimos** | "Toma de requisitos con el cliente, planificación de tareas, análisis funcional y diseño de aplicaciones, elaboración y ejecución de planes de prueba, documentación funcional [..]" |
| **Descripción de la oferta** | "Buscamos un/a diseñador/a multidisciplinar que tenga experiencia tanto en diseño web y elementos online, como en diseño offline y piezas impresas ya que tendrás que desarrollar elementos de marketing, branding y comunicación para todo tipo de soportes. [..]" |
| **Salario** | Sin especificar |
| **Fuente de datos** | InfoJobs |

Hasta ahora, se han utilizado **datos estructurados**, a pesar de que fue necesario procesarlos previamente para clasificarlos u homogeneizarlos. Pero también se utilizaron **datos no estructurados** para llevar a cabo la herramienta de análisis.


Figura 17- Clasificación de atributos según estructurados/ no estructurados

La información no estructurada se define como aquellos datos que no se encuentran contenidos en una base de datos o cualquier otra estructura de datos, por ejemplo: imágenes, audios o textos. En muchas ocasiones, la información que proporcionan los datos no estructurados es mayor a la que dan los estructurados (los datos no estructurados poseen mayor entropía), ya que estos últimos son planos y carecen de profundidad. Sin embargo, es la combinación del uso de datos tanto estructurados como no estructurados lo que potencia un mayor conocimiento de los conjuntos de datos.

Existen múltiples aplicaciones en las se utilizan técnicas de procesamiento de datos no estructurados como, por ejemplo: reconocimiento de voz, conducción autónoma o etiquetado de fotos.

Como se ha mencionado anteriormente, hay dos campos de las ofertas que son datos no estructurados: Requerimientos mínimos y Descripción de la oferta. Particularmente, se ha procesado únicamente el campo de Requerimientos mínimos puesto que es aquel que iba a ser utilizado posteriormente en la etapa de Visualización.

El objetivo de procesar el campo 'Requerimientos mínimos' era crear un gráfico de tipo nube de palabras o *WordCloud*.

**WordCloud** es una representación visual de las palabras que conforman dichos requerimientos, donde el tamaño es mayor para las palabras que aparecen con mayor frecuencia. A partir de la figura 18, podemos afirmar que la palabra "social" es aquella que aparece con mayor frecuencia, mientras "widget" es una de las palabras con menos apariciones en el texto fuente.


Figura 18- Ejemplo de WordCloud de palabras relacionadas con marketing digital. Fuente: pixabay.com

Para poder obtener la nube de palabras a partir de los Requerimientos mínimos de las ofertas, era necesario procesar cada una de sus palabras, para lo cual era necesario aplicar Procesamiento de Lenguaje Natural.

### ❖ Procesamiento de Lenguaje Natural (PLN)

El procesamiento de lenguaje natural (PLN) [29] es el conjunto de herramientas con habilidad para analizar, generar o entender el lenguaje humano. Entre las aplicaciones más importantes del PLN se encuentran:

Tabla 5- Aplicaciones de Procesamiento de Lenguaje Natural

| APLICACIÓN | DESCRIPCIÓN |
|---|---|
| **Detector automático de análisis del sentimiento** | Análisis de opinión (positiva/negativa), análisis de emoción, intención o concienciación del usuario. |
| **Extracción de tópicos** | Detección automática de las ideas de un |

| | |
|---|---|
| | texto mediante la extracción de sus temas más representativos y relevantes. |
| **Detector de reconocimiento de nombres de entidades (NER)** | Identificación de entidades completas como personas o marcas. |
| **Detector de similitudes en documentos** | Muestra el nivel de semejanza entre documentos haciendo uso de criterios semánticos y PLN. |
| **Traducción automática** | Se traduce automáticamente un texto de un idioma a otro. |

Para poder obtener un WordCloud de un texto determinado es necesario tener un conteo de las apariciones de cada palabra en el texto. Sin embargo, no todas las palabras que conforman el texto son relevantes, a pesar de que su frecuencia de aparición sea alta, como es el caso de preposiciones o conectores.

Como solución a lo anterior, el modelo utilizado ha sido "Bolsa de Palabras" o BoW, que utiliza PLN para obtener representaciones vectoriales de textos modelados con algoritmos de aprendizaje máquina.

En concreto, se ha utilizado el método de BoW para representar las palabras más utilizadas en el campo de los requerimientos mínimos que aparecen en las ofertas de empleo. Así, se podrá observar fácil e intuitivamente cuáles son los conceptos, tecnologías o aptitudes que más se solicitan en las ofertas de empleo por parte de las empresas.

Para que este modelado sea posible se necesita: un vocabulario de palabras previamente conocido y la medida de la presencia de las palabras conocidas en el texto (conteo de apariciones).

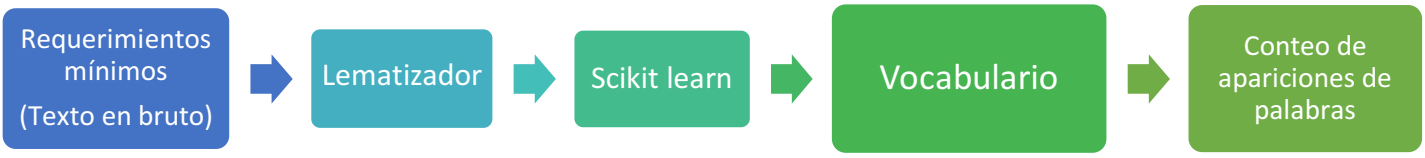



Figura 19- Procesos para generar el BoW

Como se puede observar en la Figura 11, para determinar tanto el vocabulario como el conteo de apariciones de las palabras, se hace uso de dos librerías:

- **Feature_extraction.text.CountVectorizer**.

  Es un submódulo de la librería scikit-learn. El uso de este submódulo permite convertir una colección de textos en una matriz de conteos de palabras. En este caso, dicha colección de textos es la colección de **mínimos requerimientos** de las ofertas.

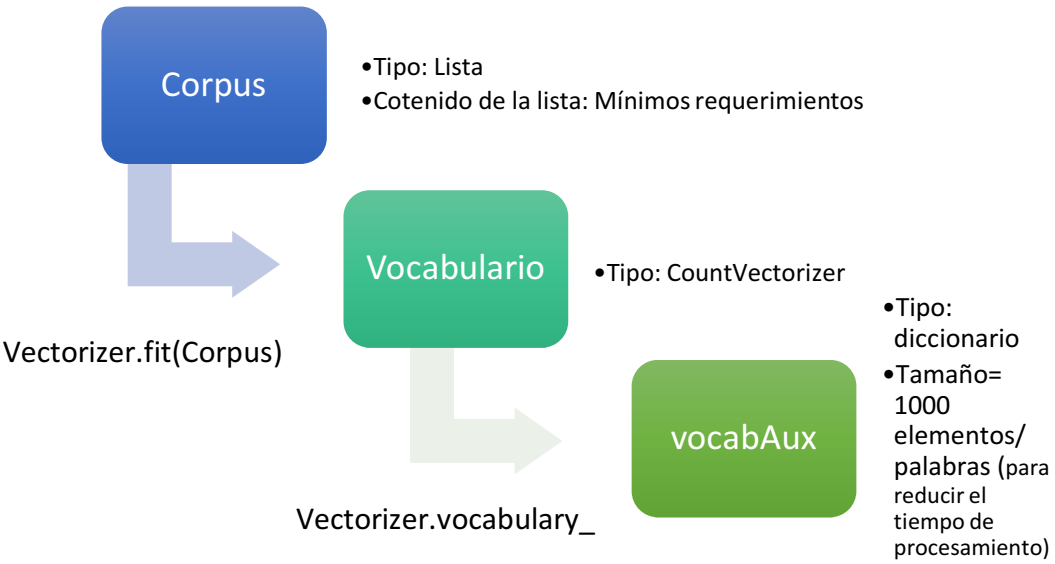
- **Lematizador**:

  La lematización es un proceso que consiste en hallar el lema de una palabra, el cual no tiene por qué tener significado. Así, un algoritmo de lematización o lematizador simplifica o normaliza palabras que aparecen en diferentes formas a una única forma común.

  El lematizador utilizado en este proyecto no es puro, pues no se queda con el lema de las palabras, ya que existiría el riesgo de que lo representado en la nube de palabras no tuviera sentido. Por esa razón, el lematizador utilizado homogeneiza las palabras eliminando plurales, género o tiempos verbales, de forma que simplemente se queda con aquella palabra representativa de todas las demás. Por ejemplo: se bloquean todas las posibles formas del verbo "tener" como: tengo, tienes, tiene, tengáis, tendríamos, tenían, tuve, etc. En este caso la palabra que permanecería para ser incluida en el vocabulario sería "tener".
  Gracias al uso del lematizador se logra reducir considerablemente las palabras innecesarias o redundantes al crear el vocabulario de palabras.

---

El proceso llevado a cabo para la elaboración del BoW es el siguiente:



1. Para empezar, se parte del texto del cual queremos obtener el *Bag of Words*: los mínimos requerimientos de las ofertas. Así, se crea la lista (corpus) cuyos elementos serán las palabras de todos los mínimos requerimientos de cada oferta.

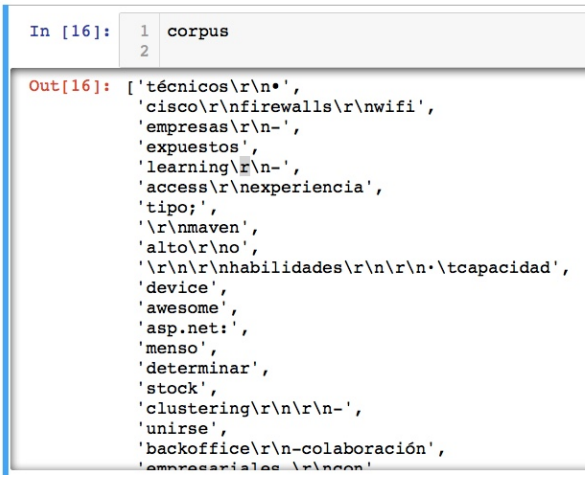



Figura 20- Lista de las palabras del campo "Requerimientos mínimos" de una oferta

2. A partir de esta lista será generado el vocabulario, ordenado alfabéticamente, utilizado como base para calcular el número de apariciones de las palabras de los mínimos requerimientos:

   - Se crea el objeto CountVectorizer(). En base a este objeto, se determinará el recuento de las veces que una palabra aparece en un texto (corpus, en este caso).
     En la figura 21 se pueden observar las palabras que contiene el objeto CountVectorizer.

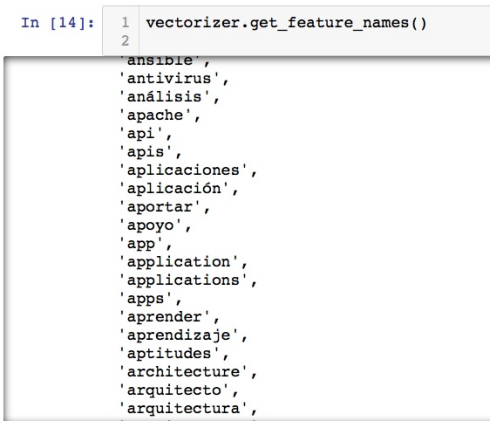



Figura 21- Palabras que conforman el vocabulario

- Tras pasar por el **lematizador**, se calculan los vectores de cada oferta que contienen el número de apariciones de una palabra en la posición del vector que ocupa dicha palabra en el vocabulario.
- Se determina cuáles son las todas las palabras (tokens) del corpus correspondiente a todas las ofertas y se les asigna un índice a cada una (se utiliza la función *fit)*, dando como resultado el vocabulario.

```
In [5]:  1 vocabulario

Out[5]: CountVectorizer(analyzer='word', binary=False, decode_error='strict',
                        dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',
                        lowercase=True, max_df=1.0, max_features=1000, min_df=1,
                        ngram_range=(1, 1), preprocessor=None, stop_words='english',
                        strip_accents=None, token_pattern='(?u)\\b\\w\\w+\\b',
                        tokenizer=None, vocabulary=None)
```

Figura 22- Vocabulario y sus atributos

Tras este proceso se obtiene una matriz dispersa de la forma: [n_posición, n_apariciones]. Donde *n_posición* hace referencia a la posición de una palabra en el vocabulario y *n_apariciones* la cantidad de veces que aparece la palabra en el conjunto de todas las ofertas.

3. El resultado de calcular este conteo de apariciones, se introduce en el JSON que se envía al JavaScript (a la fase de Visualización), transmitiendo la información de la siguiente forma:

```
Oferta_sample["minReqConteos"]=n_apariciones
Oferta_sample ["minReqPosiciones"]=n_posiciones
```

4. Por último, para exportar el vocabulario en forma de diccionario, utilizamos uno de los atributos de este módulo: *vocabulary__*, el cual permite obtener un vocabulario más legible del tipo:

```
In [7]:  1 vocabAux

Out[7]: {'técnicos': 928,
         'cisco': 182,
         'firewalls': 371,
         'wifi': 982,
         'empresas': 320,
         'learning': 520,
         'access': 35,
         'experiencia': 357,
         'maven': 551,
         'alto': 64,
         'habilidades': 415,
         'capacidad': 163,
         'asp': 99,
         'net': 614,
         'colaboración': 194,
         'empresariales': 319,
         '2016': 18,
         'competencias': 201,
         'comunicativa': 209,
         'idiomas': 439
```

Figura 23- Vocabulario final ('palabra': posición en la lista)

Este vocabulario se introduce también en el JSON que se utiliza en la parte de visualización para poder manejar adecuadamente las variaciones de la cantidad de ofertas con las que se generan los gráficos. Así, si se tienen más o menos ofertas (al filtrar) de las que representar sus mínimos requerimientos en el *WordCloud*, se puede recalcular el número de apariciones de las palabras en base a este vocabulario.

### 4.3.3 Visualización

Esta es la última parte del proceso, en la cual se pasa de tener datos sin un verdadero sentido a tener información apta para ser interpretada fácilmente y analizada. El proceso en particular puede resumirse de esta forma: se recibe un fichero JSON, cuyo contenido se precisa a continuación. Después, se generan los gráficos haciendo uso de las herramientas dc.js y Crossfilter (construidas sobre d3.js).

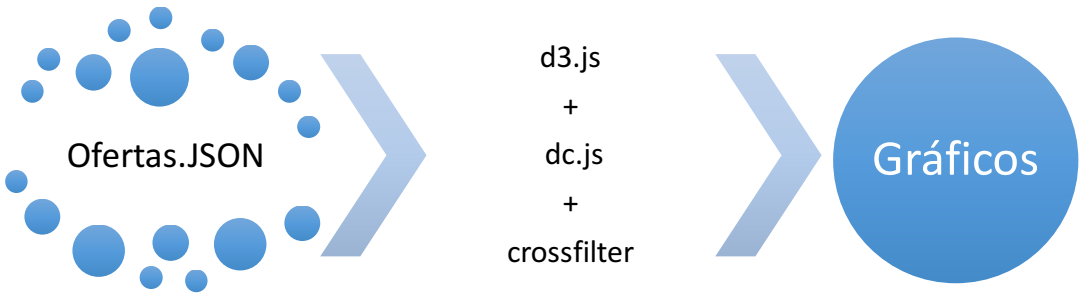



Figura 24- Esquema del proceso de transformar datos a gráficos

En esta parte del proyecto se obtienen los resultados gráficos a partir de los datos que provienen del archivo JSON, "Ofertas.JSON", formado por las siguientes partes:

En la tabla 6 se muestra el contenido del archivo en formato JSON que recibe esta fase del proyecto.

Tabla 6- Contenido de las partes del archivo "Ofertas.json"

| NOMBRE | CONTENIDO |
|---|---|
| **Data** | Contiene todas las ofertas de empleo ya procesadas. Es decir, en formato JSON, con cada uno de sus atributos diferenciados y sus respectivos valores. |
| **GeoJson** | Contiene la representación (coordenadas) de las provincias de España, junto con sus atributos como: nombre de la provincia y su código identificador. |
| **Vocabulary** | Contiene el vocabulario ordenado alfabéticamente con las palabras que componen los mínimos requerimientos que aparecen en las ofertas. |

Relativo al proceso de creación de la herramienta de visualización, fue necesario determinar qué gráficos serían útiles y efectivos para conseguir los objetivos expuestos al comienzo de la presente memoria.

Como se explicó en el desarrollo de las herramientas utilizadas, el uso de las librerías d3.js y dc.js facilita la asociación de los diferentes gráficos a las partes del DOM diseñado y su variación en función del flujo de datos.

Por otro lado, el uso de la librería **Crossfilter** permite examinar grandes conjuntos de datos y permite la rápida interacción con vistas coordinadas. Esta fluidez se debe a que solo se construye la vista de cero una única vez, de forma que cualquier interacción supone simplemente pequeños ajustes en los filtros, filtrado incremental y reducción, como se explicó en apartados anteriores. En este caso particular, se utilizan las funciones de add(), reduce() e initialize() para manejar las variaciones interactivas o filtros cruzados relativos a los requerimientos de las ofertas, como se explicará en próximos apartados.

En la tabla 7, se exponen cuáles son los campos de las ofertas de empleo que se consideraron más relevantes para ser representados y con qué tipo de gráfico se hace en cada caso.

Tabla 7- Tipo de gráfico escogido para cada atributo de una oferta

| Datos que representa | Tipo de gráfico | Nombre del gráfico en dc.js |
|---|---|---|
| **Fuente de datos** | Gráfico de sectores | pieChart |
| **Experiencia mínima requerida** | Gráfico de barras | barChart |
| **Categorías** | Gráfico de sectores | pieChart |
| **Salario** | Gráfico de sectores | pieChart |
| **Distribución de ofertas en España** | Mapa | mapChartProv |
| **Mínimos requerimientos** | Word Cloud | -No pertenece a la librería dc.js- |
| **Fecha de creación** | Gráfico de barras<br>+<br>Gráfico de líneas | barChart<br>+<br>lineChart |

En la figura 25, se presenta la plantilla seguida para el diseño estructural de la página web que contiene la herramienta solución. Se trata de un boceto final, tras haber sido valoradas diversas opciones. Para llegar a dicha versión, se tuvo en cuenta la relevancia de cada gráfico, así como el tamaño apropiado para cada uno de ellos.

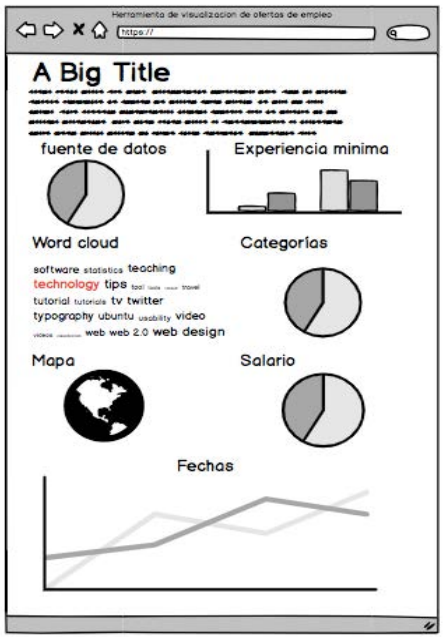



Figura 25-Esquema organización de los gráficos de la página web

A continuación, se van a presentar cada una de figuras de la herramienta:

1. **Gráfico de sectores- Fuente de datos**: permite seleccionar la fuente a la que pertenecen las ofertas.
   - Dimensión: campo “fuente” de las ofertas de empleo.
   - Grupo: agrupación de los valores del campo “fuente”.
   - Tipo de gráfico dc.js: pieChart


Figura 26- Gráfico de sectores con fuente de datos

2. **Gráfico de sectores- Categorías:** permite filtrar las ofertas según la **categoría** a la que pertenecen.
   - Dimensión: campo “Category” de las ofertas de empleo.
   - Grupo: agrupación de aquellas ofertas de la misma categoría
   - Tipo de gráfico dc.js: pieChart


Figura 27- Gráfico de sectores con categorías

Como se puede observar, un 93% aproximado de las ofertas sin aplicar ningún filtro son de la categoría *Informática y telecomunicaciones*, ya que ésta es la categoría de todas las ofertas de Tecnoempleo.

3. **Gráfico de barras- Experiencia mínima requerida:** permite filtrar las ofertas según la experiencia mínima que requieren las empresas.
    - Dimensión: campo "experiencia mínima" de las ofertas de empleo
    - Grupo: agrupa aquellas ofertas con la misma experiencia mínima requerida.
    - Tipo de gráfico dc.js: barChart

    En este caso, fue necesario homogeneizar los campos en ambas fuentes de datos, ya que aparecían definidas de diferente manera y por ello era imposible agruparlas.


Figura 28-Gráfico de barras con experiencia mínima

4. **Gráfico de sectores – salario:** permite al usuario filtrar las ofertas por el salario ofrecido.
    - Dimensión: campo de "salario" indicado en las ofertas.
    - Grupo: agrupación de las ofertas con mismo salario mínimo ofrecido.
    - Tipo de gráfico dc.js: pieChart

Esta información solo es completada por las ofertas provenientes de Tecnoempleo, por lo que, si se filtra de forma que solo aparezcan las ofertas de InfoJobs, este gráfico quedará en un 100% del valor "sin especificar".


Figura 29-Gráfico de sectores con salarios

**5. Gráfico de líneas/barras- Fechas de creación:** permite ver la evolución temporal de la cantidad de ofertas de empleo subidas a los portales web. Este gráfico no es seleccionable de forma que no filtra las ofertas, sin embargo, permite interacción del usuario ya que permite desplazarse por la gráfica a lo largo del eje temporal, pudiendo ampliar el eje de tiempos para un enfoque más concentrado.

- Dimensión: fecha de creación de las ofertas.
- Grupo: suma de aquellas ofertas creadas en el mismo día.
- Tipo de gráfico: lineChart y barChart.


Figura 30- Gráfico de fechas de creación de ofertas (ampliable)

**6. Mapa de España:** permite seleccionar las diferentes provincias por las cuales se desea filtrar las ofertas.

El tipo de gráfico de dc.js utilizado es *dc.geoChoroplethChart.* Este gráfico permite crear un mapa a partir de datos en un geoJSON, manejado por un filtro cruzado.

GeoJSON [30] es un formato basado en JSON específicamente diseñado para codificar una variedad de estructuras de datos geoespaciales. En este caso se ha obtenido el fichero GeoJSON a través de la herramienta que ofrece www.geojson.io, donde se puede crear de una forma muy visual y accesible el GeoJSON la región que se elija, en este caso, España.

Dentro el archivo de tipo GeoJSON, *FeatureCollection* es la región comprendida por cada *Feature* (objetos geométricos que conforman cada región), además cada *Feature* tiene unas coordenadas y un ID, para que pueda ser identificada cada provincia dentro del mapa. En el siguiente fragmento se puede observar un ejemplo de un "*Feature*" correspondiente a la provincia Girona, con el código de provincia (ID) 17.

{"type":"Feature","geometry":{"type":"MultiPolygon","coordinates":[[[[-13.890506,28.756853],[-13.836324,28.712746],[-13.824298,28.552822],[-13.867233,28.480298],[-13.849884,28.401132],[-13.917353,28.25195],[-14.012607,28.207694],[-14.218269,28.165583],[-14.33159,28.044622],[-14.477795,28.07747],[-14.507192,28.064071],[-14.491428,28.110908],[-14.450234,28.100089],[-14.372972,28.117134],[-14.222542,28.215121],[-14.203695,28.328393],[-14.162317,28.368689],[-14.147239,28.438396],[-14.100506,28.475284],[-14.033462,28.592999],[-14.038761,28.615067],[-¡15.406981,28.158374],[…],"properties":{"cod_prov":"17","name":"Girona","cod_ccaa":"08","cartodb_id":17,"created_at":"2014-09-30T00:00:00Z","updated_at":"2014-12-25T01:56:10Z"}}

En la figura 31, se ha realizado un ejemplo de creación del fichero geoJSON en el cual se seleccionan los puntos de la frontera del país, sin embargo, en el presente proyecto se han seleccionaron todas las provincias en su extensión.

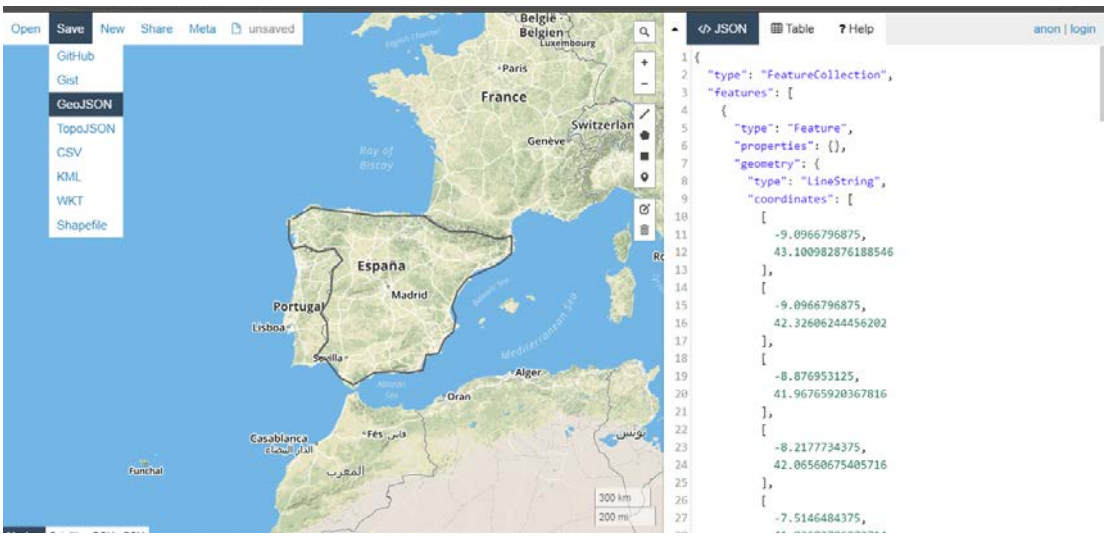



Figura 31- Ejemplo de creación de un geoJson mediante www.geojson.io

Una vez se obtuvo la correcta representación del mapa en la herramienta diseñada (ver figura 32), se tuvieron que mapear los códigos de las provincias utilizados en las ofertas para que se correspondieran con los IDs de las provincias del geoJSON. Para ello fueron necesarios tres diccionarios, cuyos fragmentso se muestran a continuación, ya que había que homogeneizar los códigos de provincias de Tecnoempleo e InfoJobs y después, mapearlos a los IDs del geoJSON.


Figura 32- Mapa de distribución de ofertas por provincias

```
dict_prov_IJ["5"]="04";
dict_prov_IJ["36"]="30";
dict_prov_IJ["3"]="02";
dict_prov_IJ["7"]="05";
dict_prov_IJ["2"]="01";
dict_prov_IJ["8"]="06";
dict_prov_IJ["4"]="03";
dict_prov_IJ["38"]="32";
```

Figura 33- Fragmento del diccionario para homogene izar los códigos de las provincias de ofertas de InfoJobs

```
dict_prov_TE["Murcia"]="30";
dict_prov_TE["Albacete"]="02";
dict_prov_TE["Ávila"]="05";
dict_prov_TE["Álaba"]="01";
dict_prov_TE["Badajoz"]="06";
dict_prov_TE["Alicante"]="03";
dict_prov_TE["Ourense"]="32";
dict_prov_TE["Barcelona"]="08";
dict_prov_TE["Burgos"]="09";
dict_prov_TE["Cáceres"]="10";
dict_prov_TE["Cádiz"]="11";
dict_prov_TE["Castellón"]="12";
dict_prov_TE["Ciudad real"]="13";
dict_prov_TE["Jaén"]="23";
dict_prov_TE["Córdoba"]="14";
dict_prov_TE["Cuenca"]="16";
```

Figura 34-Fragmento del diccionario para homogeneizar los códigos de las provincias de ofertas de Tecnoempleo

```
diccionario_provincias.Asturias:{cod_v:33, cod_m:6, name: "Asturias"};
diccionario_provincias.IllesBalears:{cod_v:07, cod_m:26, name: "Illes Balears"};
diccionario_provincias.Acoruña:{cod_v:15, cod_m:28, name: "A Coru\xf1a"};
diccionario_provincias.Girona:{cod_v:17, cod_m:19, name: "Girona"};
diccionario_provincias.LasPalmas:{cod_v:35, cod_m:20, name: "Las Palmas"};
diccionario_provincias.40:{cod_v:36, cod_m:40, name: "Pontevedra"};
diccionario_provincias.SantaCruz:{cod_v:38, cod_m:46, name: "Santa Cruz de Tenerife"};
diccionario_provincias.Cantabria:{cod_v:39, cod_m:13, name: "Cantabria"};
diccionario_provincias.Malaga:{cod_v:29, cod_m:34, name: "M\xe1laga"};
diccionario_provincias.Almeria:{cod_v:04 ,cod_m:5, name: "Almer\xeda"};
diccionario_provincias.Murcia:{cod_v:30 ,cod_m:36, name: "Murcia"};
diccionario_provincias.Albacete:{cod_v:02,cod_m:3, name: "Albacete"};
diccionario_provincias.Avila:{cod_v:05, cod_m:7, name: "\xc1vila"};
diccionario_provincias.Alaba:{cod_v:01,cod_m:2, name: "\xc1lava/Araba"};
diccionario_provincias.Badajoz:{cod_v:06, cod_m:8, name: "Badajoz"};
diccionario_provincias.Albacete:{cod_v:03, cod_m:4, name: "Alicante/Alacant"};
```

Figura 35-Fragmento del diccionario para homogeneizar el código del geoJson con el de las ofertas

**7.Word Cloud:** nube de palabras con los mínimos requerimientos.

Para generar el Word Cloud o nube de palabras se utiliza un script disponible en GitHub con licencia de código abierto creado por Jason Davies [31]
Existe una herramienta online que automáticamente calcula el número de apariciones de las palabras en un texto dado y proporciona su Word Cloud correspondiente. En este caso ha sido preciso utilizar el script que genera el gráfico a partir de una lista tipo: [“palabra”,” numero de apariciones”].

**Funcionamiento del algoritmo:**

Tomando en primer lugar las palabras de mayor ocurrencia, ésta se coloca cerca del medio, requiriendo mayor atención visual. Si al añadir una nueva palabra esta se cruza con alguna colocada anteriormente, se mueve un “paso” a lo largo de una espiral creciente [32].

Al no pertenecer este gráfico a la librería dc.js, fue pertinente implementar las funciones de “add”, “reduce” e “initialize”, que manejan los cambios del gráfico cuando se añaden nuevas palabras en la bolsa de palabras, se eliminan o se crea, respectivamente. Estas variaciones se producen cuando el usuario realiza un filtro sobre cualquier otra gráfica de la herramienta y se reciben mediante el evento de tipo On.

A continuación, se puede ver la implementación de la función “add” que maneja el aumento de número de ofertas en el filtrado:

```
        function(conteoBow, d){
                        console.log("add");
                        d.minReqPosiciones.forEach(function(pos,index){
                                conteoBow[pos] +=d.minReqConteos[index];
                                });
                        return conteoBow;
        },
```

En la figura 36 se puede observar aquellas palabras con mayor frecuencia de aparición en el atributo de “Mínimos requerimientos” del conjunto de ofertas de empleo seleccionadas por el usuario.

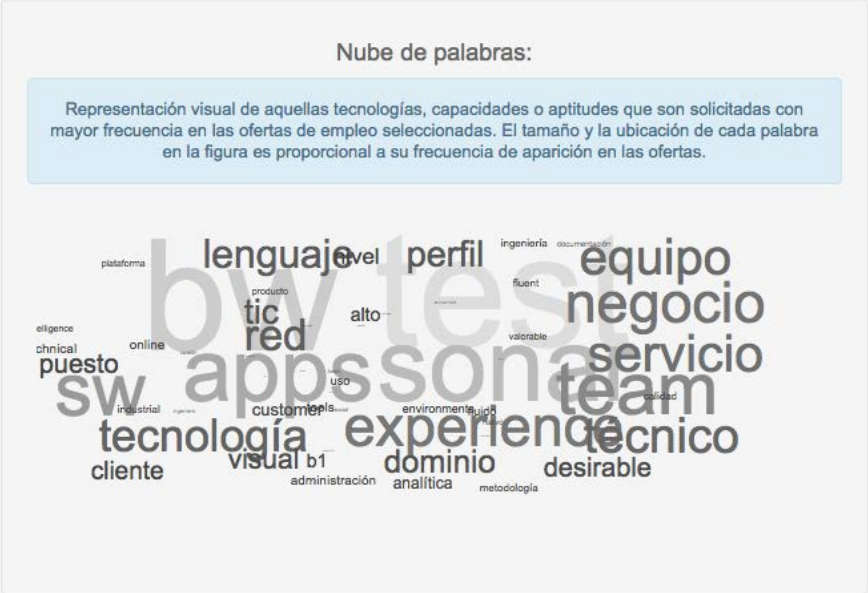



Figura 36- Nube de palabras

# 5-RESULTADOS

Tras el proceso de diseño e implementación de la solución propuesta en el anterior capítulo, se llegó a la versión final de la herramienta, a falta de evaluación y posibles cambios de la misma.

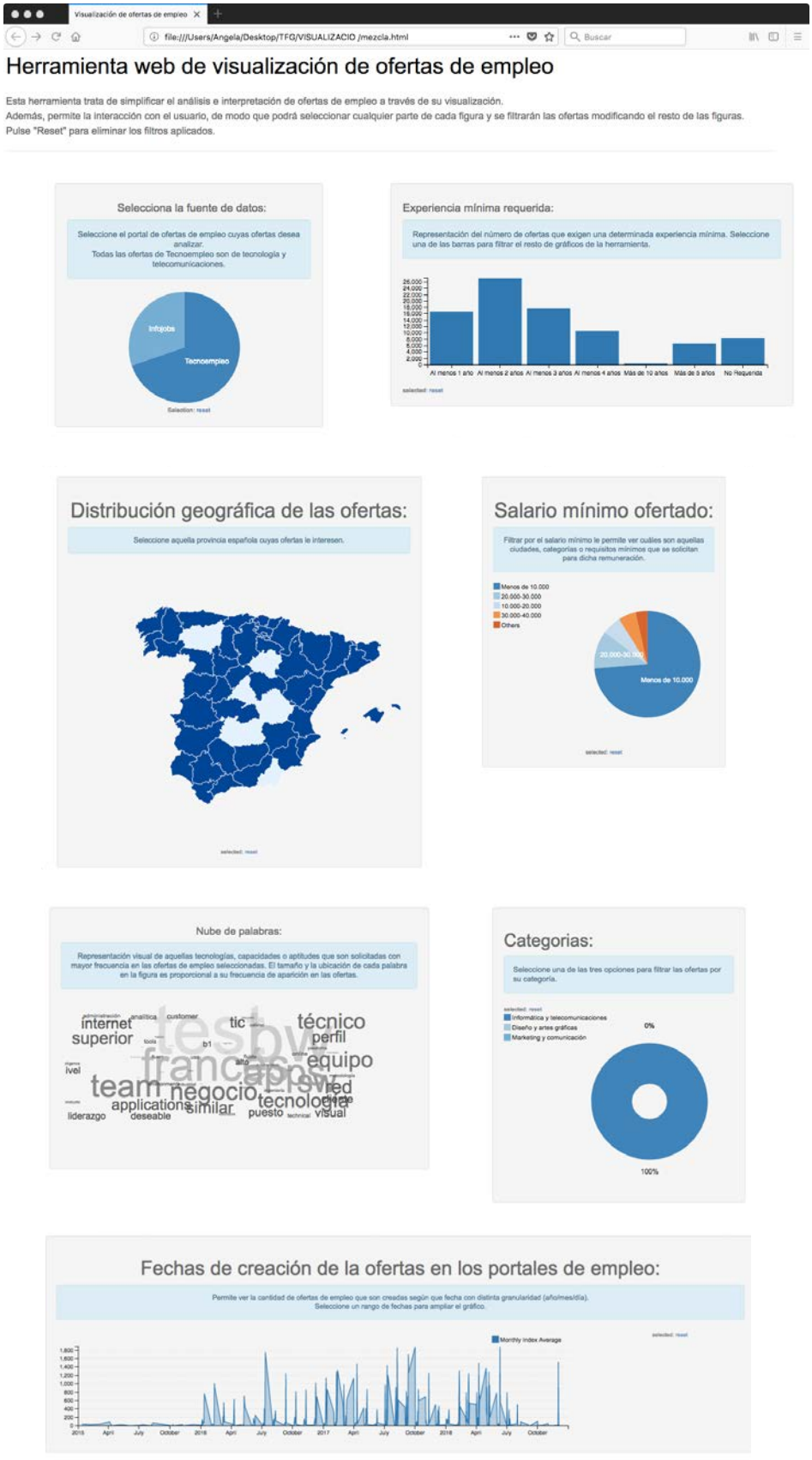



Figura 37- Resultado final de la herramienta antes de la evaluación

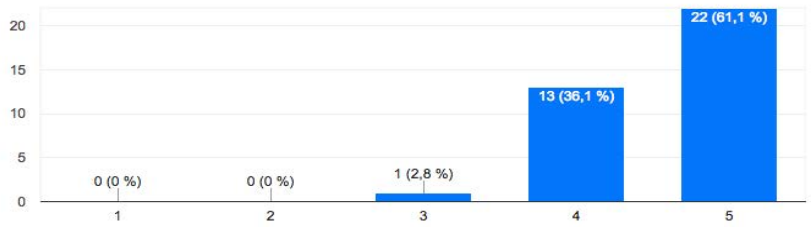
# 6- EVALUACIÓN

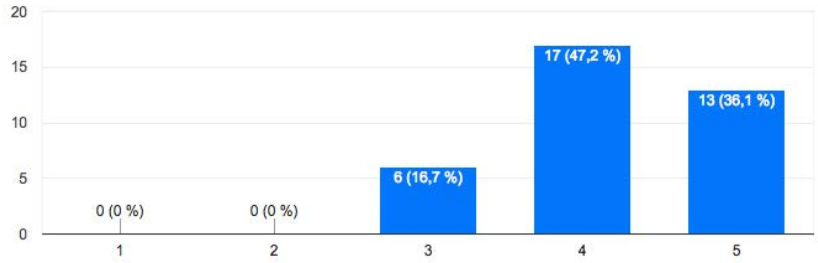
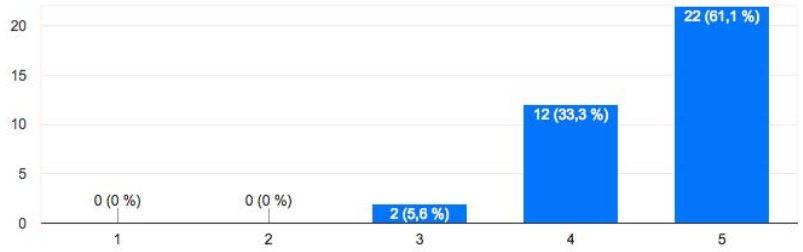
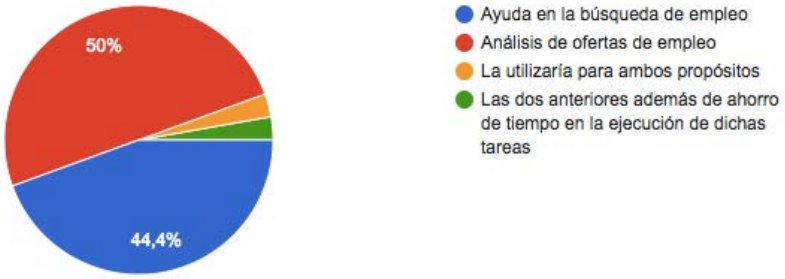
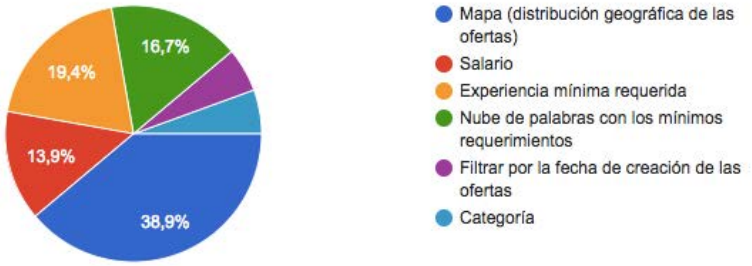
Tras obtener la herramienta web anterior, fue sometida a examen mediante distintas pruebas para su posterior mejora:

- En primer lugar, se estudió el tiempo que llevaba la carga y actualización de los datos en la herramienta. El tiempo superaba lo deseado, con lo que se redujeron el número de palabras a procesar en la nube de palabras; ya que una gran cantidad de palabras aumentaba considerablemente el tiempo de procesado y entorpecían su visualización.
- Por otra parte, la estructura de la página web se había realizado mediante tablas, donde cada gráfico se introducía dentro de una celda y así consecutivamente. Sin embargo, se decidió cambiar al uso del *framework Bootstrap*, el cual facilita la aplicación de estilos.
- La parte más aleccionadora de esta fase de "Evaluación" fue la realización de una **encuesta** donde se obtuvo una gran respuesta en forma de críticas constructivas y posibles mejoras por parte de los usuarios.
  Debido a que uno de los objetivos principales era desarrollar una herramienta útil, se consideró oportuno estudiar si se había cumplido dicho objetivo mediante experimentos reales con distintos usuarios.

  La encuesta completada por los usuarios se puede observar en el Anexo de la presente memoria, ante dichas cuestiones se obtuvieron determinadas tendencias que merecían ser consideradas para mejoras de la herramienta y posibles trabajos futuros.

Tabla 8- Resultados de la encuesta a usuarios

| | |
|---|---|
| Valoración global de la herramienta |  |

| | |
|---|---|
| Valoración de la interfaz |  |
| Valoración de la utilidad de la herramienta |  |
| Uso de la herramienta |  |
| Gráficos más útiles |  |

A parte de las conclusiones que se pueden obtener de estos gráficos generados a partir de las respuestas, lo más concluyente de ella fue que un **75%** de los encuestados respondieron que **sí** utilizarían esta herramienta como complemento a la búsqueda de empleo. Este resultado se encuentra estrechamente relacionado con que la mayoría de los encuestados son estudiantes.

A continuación, se van a exponer aquellas recomendaciones más mencionadas en el apartado de “Mejoras de la herramienta”, las cuales se tuvieron en cuenta para realizar correcciones y mejoras que permitieran concluir este proyecto con la versión más óptima posible, además de orientada al usuario.

- Mejora de la interfaz, uso de una mayor variedad de colores, adaptación al tamaño de la pantalla, etc.
- Mayor fluidez de la herramienta.
- App para el móvil / habilitar vista para el móvil.
- Guardar los filtros aplicados.
- Eliminar palabras sin sentido de la nube de palabras.

# 7-PLANIFICACIÓN Y PRESUPUESTO

## 7.1- Planificación

En el presente capítulo se detallará la duración de las tareas completadas en cada fase del proyecto. También se mostrarán las hojas de cálculo utilizadas durante el proceso de creación del proyecto, en las cuales se precisaron las tareas a realizar durante cada etapa y el grado de cumplimiento de dichas tareas durante las mismas, utilizando para ello el siguiente código de colores: **0%- 50%- 75%- 100%.** En estas hojas de cálculo las horas de trabajo eran orientativas y estimadas, pues iban variando a medida que el proyecto iba avanzando.

El tiempo necesario para el desarrollo de este trabajo de fin de grado ha sido de 8 meses, entre febrero de 2018 y septiembre de 2018, tiempo durante el cual se pueden distinguir tres etapas:

- **Primera etapa:** se corresponde con el inicio del proyecto, elección de los objetivos y posibles ampliaciones dependientes del trascurso del trabajo.

  En esta etapa se llevó a cabo la lectura de artículos relacionados con el uso de Python para el tratamiento de datos y la familiarización con los entornos de trabajo, como *Jupyter* de Anaconda o las herramientas de desarrollador que ofrece Firefox, como la consola o el depurador.

  La lectura del libro *Interactive Data Visualization* [21], de Scott Murray, fue el proceso de formación que más tiempo tomó, puesto que requiere una continuidad y comprensión de cada tema para poder avanzar en su lectura.

  Este libro supuso la mayor parte del aprendizaje de d3, ya que que es un tutorial modular y asequible para aquellos que parten de poco conocimiento en el ámbito de dicha librería de JavaScript.

  Puesto que d3.js es una herramienta construida sobre JavaScript, fue necesario adquirir conocimientos de JavaScript previo a abordar el libro tutorial de d3.js. Este aprendizaje se realizó mediante la lectura de artículos.

| | | ETAPA 1 : INICIO Y FORMACIÓN | |
|---|---|---|---|
| | Tarea 1 | Tarea 2 | Tarea 3 |
| Descripción | Lectura libro | Jupyter | Formación Python y JavaScript |
| Duración | 50 horas | 1 hora | 20 horas |
| Grado de cumplimiento | 80% | 100% | 75% |

Figura 38- Organización: Excel de la fase 1

- **Segunda etapa**: En esta segunda etapa se llevó a cabo el desarrollo de la herramienta de visualización, siguiendo el orden expuesto en capítulos anteriores:

Pre-procesado de datos → Visualización

Esta etapa fue la más larga al ser la más práctica y ser la que más problemas supuso. Aunque el desarrollo comenzó con el procesamiento de datos fue necesario recurrir a ello a medida que la herramienta de visualización avanzaba.

La planificación seguida en esta etapa fue: desarrollo de la herramienta de visualización junto con reuniones semanales con la tutora para la resolución de dudas y/o revisión del desarrollo.

Si bien la primera etapa fue fundamentalmente de aprendizaje, en esta etapa el aprendizaje fue intensivo y continuado, pues se trataba de resolver errores y problemas prácticos, buscando alternativas o soluciones efectivas ante ellos.

| | | ETAPA 2: PRE-PROCESADO DE DATOS | | |
|---|---|---|---|---|
| | Tarea 1 | Tarea 2 | Tarea 3 | Tarea 4 |
| Descripción | Descarga de datos | Lectura ficheros | Creación de diccionarios | Volcado de datos a JSON |
| Duración | 1 hora | 7 horas | 3 horas | 3 horas |
| Grado de cumplimiento | 100% | 100% | 75% | 100% |

| | | ETAPA 2: DESARROLLO DE LA HERRAMIENTA | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Tarea 1 | Tarea 2 | Tarea 3 | Tarea 4 | Tarea 5 | Tarea 6 | Tarea 7 | Tarea 8 | Tarea 8 |
| Descripción | Gráfico categorías | Gráfico experiencia | Gráfico fuente | Gráfico salario | Mapa | NLP | BoW | Plantilla CSS | Estructura HTML |
| Duración | 5 hora | 4 horas | 3 horas | 2 horas | 9 horas | 8 horas | 12 horas | 3 horas | 9 horas |
| Grado de cumplimiento | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Figura 39- Organización: Excel de la fase 2

- **Tercera etapa:** esta última etapa incluye la escritura de la presente memoria, así como mejoras funcionales y de diseño.
  Si bien durante la etapa anterior se llevó a cabo un registro de las actividades resueltas y problemas emergentes para la posterior realización de la memoria, ésta se dejó para el final al tratarse de un ejercicio más independiente y autónomo, sin requerir ayuda de la tutora.

| | | ETAPA 3: MEMORIA, EVALUACION Y MEJORAS | | | |
|---|---|---|---|---|---|
| | Tarea 1 | Tarea 2 | Tarea 3 | Tarea 4 | Tarea 5 |
| Descripción | Estructura memoria | Estado del arte | Escritura de la memoria | Evaluacion | Mejoras |
| Duración | 1 hora | 3 horas | 60 horas | 6 horas | 6 horas |
| Grado de cumplimiento | 100% | 100% | 50% | 5% | 30% |

Figura 40- Organización: Excel de la fase 3

Tabla 9- Tareas: descripción y duración

| Tarea | Descripción | Duración |
|---|---|---|
| **Inicio del proyecto** | Elección del tema y determinación de los bloques funcionales | 1 día |
| **Organización del proyecto** | Definición de las partes funcionales, tiempo y dificultad | 4 días |
| **Documentación** | Lectura de documentación de las tecnologías a utilizar durante el proyecto | 30 días |
| **Formación** | Cursos online y autoaprendizaje | 30 días |
| **Instalación de programas** | Instalación de anaconda y aprendizaje de uso del entorno de trabajo Jupyter notebook | 2 días |
| **Obtención de datos a utilizar** | Recopilación, selección y almacenamiento de los datos a utilizar | 10 días |
| **Pre-procesamiento** | Implementación en Python del pre-procesamiento de datos | 30 días |
| **Elección de gráficos** | Elección de los gráficos para representar determinados campos de las ofertas | 3 días |
| **Diseño del prototipo de página web** | Diseño de la interfaz, situación de cada gráfico y su tamaño | 1 día |
| **Implementación de la página web** | Estructura en HTML y creación de las hojas de estilo CSS | 5 días |
| **Implementación de los gráficos** | Implementación de los distintos gráficos haciendo uso de tecnologías basadas en JavaScript | 50 días |
| **Evaluación de la herramienta** | Testeo de la herramienta y realización de una encuesta a un grupo de usuarios | 6 días |
| **Modificaciones de la herramienta** | Mejoras y correcciones de la herramienta tras la evaluación | 15 días |
| **Nuevos datos** | Añadido de datos a la herramienta y modificación de los scripts para mejorar la fluidez | 1 día |
| **Pruebas definitivas** | Últimas pruebas y testeo del funcionamiento. Concreción de líneas futuras | 2 días |

| | | |
|---|---|---|
| **Memoria** | Redacción de la presente memoria | 80 días |

En la siguiente figura, se representa el **diagrama de Gantt**, una herramienta utilizada en la organización inicial del proyecto que marca el origen y fin de cada tarea, lo cual facilitó el control global del proyecto, aumentando la eficacia de cada tarea.

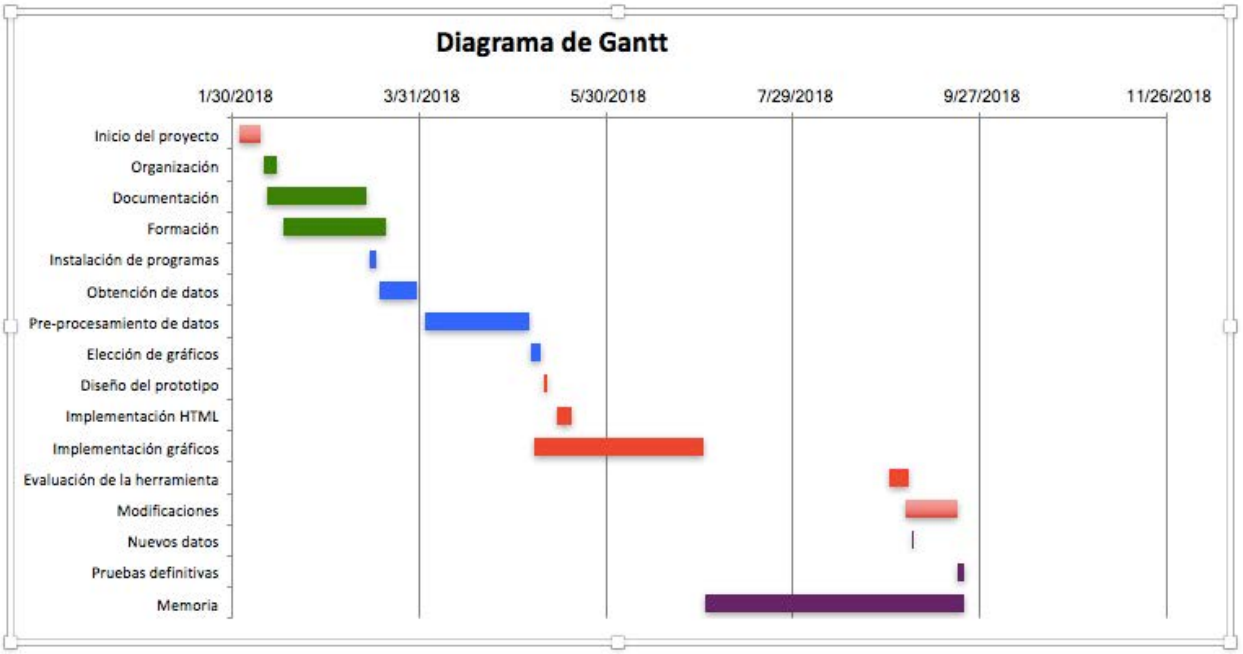



Figura 41- Diagrama de Gantt

## 7.2- Presupuesto e impacto socio-económico

### 7.2.1 Presupuesto

Es importante en todo proyecto realizar un presupuesto que contabilice tanto la mano de obra como programas (software) y medios tecnológicos (hardware) utilizados en su desarrollo.

- **Material (software + hardware)**: Para la elaboración de este trabajo se ha utilizado un ordenador durante los 8 meses de duración. Considerando la vida útil media aproximada del portátil 5 años y un coste de 1000 €, su amortización en este periodo de tiempo será:

$$\frac{8\ meses}{5\ años * 12\ meses/año} * 1000€ = 133{,}33€$$

Respecto al software, todos los programas informáticos utilizados no son considerados como gasto, pues o bien se disponía de una licencia educativa gratuita (Microsoft Office), o se han utilizado plataformas de código libre y abierto.

- **Mano de obra:** Deben tenerse en cuenta las horas dedicadas al proyecto por cada persona implicada, cuyo coste se estima en la tabla 10:

Tabla 10- Coste del personal

| Nombre personal | Categoría | Tiempo dedicado | €/h | Coste total |
|---|---|---|---|---|
| **Ángela Moreno Martínez** | Alumna | 400 h | 23,43€/h | 9372€ |
| **Vanessa Gómez Verdejo** | Tutora | 40 h | 35,33€/h | 1413,2€ |
| | | | **Coste total mano de obra:** | **10785,20 €** |

Teniendo en cuenta los costes de hardware, así como de mano de obra, el presupuesto aproximado completo sería de un total de 10918,53€.

### 7.2.2 Impacto socio-económico

Se espera un impacto tanto económico como social de la herramienta desarrollada en este proyecto.

En el ámbito **económico,** se contempla la opción de comercializar esta herramienta a plataformas web de búsqueda de empleo como InfoJobs o Tecnoempleo, cuya aplicación sería directa ya que se explotan sus propios datos. La posibilidad del usuario de visualizar e interactuar con las distintas figuras que representan y filtran las ofertas de empleo puede mejorar y aumentar la eficiencia y consistencia de dichas plataformas.

En el ámbito **social,** se tienen en cuenta los resultados de la encuesta realizada en este proyecto. En ella, la mayoría de los encuestados han admitido sentirse interesados por el uso de esta herramienta en situaciones de búsqueda de empleo; además de recomendar un futuro desarrollo de la herramienta para poder utilizarla en sus smartphones o como parte integrada de otras páginas al uso como InfoJobs o Tecnoempleo.

Como conclusión, si bien esta herramienta por sí sola no es útil para encontrar empleo, sí lo es en combinación con otras herramientas. Dicha combinación tiene cabida en este sector, donde Internet es la herramienta fundamental para la búsqueda de empleo.

# 8.CONCLUSIONES Y TRABAJO FUTURO

## 8.1 Conclusiones

En un mundo laboral cada vez más competitivo, nuevos puestos de trabajo surgen debido a la transformación digital del mercado en todas sus vertientes. La automatización y digitalización de los procesos en las empresas provoca un aumento de la cantidad de profesionales tecnológicos en las plantillas, aquellos capaces de aportar valor en todas aquellas metodologías o tecnologías que forman parte del negocio.

Esta situación ha provocado una brecha entre las empresas y los profesionales, la cual se resuelve mediante una mayor difusión de información sobre cuáles son los nuevos requisitos laborales que las empresas exigen actualmente [28]. Tal y como se propuso en el comienzo de esta memoria, uno de los objetivos que pretendía alcanzar la herramienta de visualización presentada era disminuir esta brecha entre empresas y profesionales.

Como se puede apreciar en el resultado de la evaluación, la herramienta web de visualización de ofertas de empleo ha satisfecho este objetivo proporcionando una interfaz cómoda y sencilla para que los profesionales sean conscientes de cuáles son las competencias exigidas, así como los puestos de trabajo ofertados por empresas reconocidas en varios sectores.

## 8.2 Trabajo futuro

Este trabajo puede servir de base para una herramienta web más elaborada, donde la variabilidad de los gráficos sea mayor o incluso se utilicen nuevas librerías que mejoren la interfaz de le herramienta, por ejemplo, mediante el uso de **gráficos en 3D**.

Una importante mejora a realizar sería **almacenar los datos descargados** y ya pre-procesados de las ofertas de empleo en una base de datos para que la herramienta fuese accesible desde cualquier dispositivo. Así, la información se almacenaría en forma de tablas y los datos serían obtenidos por la herramienta mediante consultas a dichas tablas. SQL sería el lenguaje utilizado para administrar dicha base de datos. Esto no se hizo durante este proyecto puesto que el tratamiento y administración de bases de datos no era uno de los objetivos a alcanzar, pues ya se poseen conocimientos sobre esta tarea y no se consideró que aportara un verdadero aprendizaje. En este proyecto se pretendió focalizar el aprendizaje en el procesamiento de los datos y su visualización más que en la parte del almacenamiento y organización de los mismos.

El uso de **otras bolsas de empleo** aumentaría la significación de los datos, por ejemplo, aumentando las categorías de empleos, puesto que actualmente la herramienta solo incluye empleos pertenecientes a los sectores de: Informática y Telecomunicaciones, Marketing y comunicación y Artes gráficas.

Otra de las posibles funciones futuras sería hacer posibles analíticas en función del uso de la página, esto es: almacenar hábitos de navegación y otro tipo de información que pueda ser útil en el ámbito del marketing digital. En función de estos hábitos de navegación, podría crearse un sistema de recomendación, donde aparezcan ofertas orientadas al gusto del usuario, lo cual se denomina ***clustering***.

Una interesante línea para ampliar este proyecto sería introducir en la herramienta un "**análisis de tópicos".** Este es un caso de aprendizaje no supervisado, donde para unos determinados datos de entrada, no se tienen unos determinados datos de salida posibles (no existen datos etiquetados). De esta forma, se estudiaría la estructura de los datos para encontrar posibles etiquetas o grupos bajo los cuales se puedan agrupar o clasificar un conjunto de los datos. Por ejemplo: se podría encontrar que las ofertas en el este de Madrid piden un mayor nivel de inglés que el resto (sería un nuevo tópico que comparte un conjunto de los datos).

Otra línea de futuro a considerar, la cual fue altamente recomendada en la encuesta de evaluación es obtener una versión de la herramienta para ***smartphones***, o desarrollar una app.

# BIBLIOGRAFÍA


[1] "¿Cómo afectará la transformación digital y la robotización al empleo en las pymes*?", Tecnología para los negocios*. [En línea]. Accedido en 21/8/18. Disponible en: https://ticnegocios.camaravalencia.com/servicios/tendencias/como-afectara-la-transformacion-digital-y-la-robotizacion-al-empleo-en-las-pymes/ [Accedido: 15-08-2018]

[2] M.J.A. Berry, G.S. Linoff. *Data mining Techniques*, third edition. United States of America: John Wiley & Sons, 1997.

[3] D,Pyle. *Data preparation for data mining.* United States of America: Morgan Kauffman: 1999, páginas 90-95.

[4] J.Hernández. *Introducción a la minería de datos, 1ª edición. España: Pearson Educacion:2004, 12.*

[5] "Visualización de datos: definición, tecnologías y herramientas.", *Red.es.* [En línea]. Disponible en: http://datos.gob.es/sites/default/files/doc/file/informe_herramientas_visualizacion.pdf [Accedido: 23-6-2018]

[6] "Tecnoempleo.com", *Tecnoempleo* [En línea]. Disponible en: https://www.tecnoempleo.com [Accedido: 21-6-2018]

[7] "InfoJobs- Bolsa de trabajo, ofertas de empleo", *InfoJobs.* [En línea]. Disponible en: *https://www.infojobs.net/* [Accedido: 21-6-2018]

[8] J.Mendiola. "Cómo encontrar trabajo aprovechando las redes sociales"*, El Pais*, 17/02/2018. [En línea]. Disponible en: https://elpais.com/tecnologia/2018/02/15/actualidad/1518712262_700981.html [Accedido: 24-6-2018]

[9] C.Hidalgo. "¿Para qué sirve LinkedIn?", *El Pais*, 25/07/2018. [En línea]. Disponible en: https://elpais.com/economia/2018/07/12/actualidad/1531414339_328165.html [Accedido: 24-6-2018]

[10] "Introducción a JSON", *json.* [En línea]. Disponible en: https://www.json.org/json-es.html [Accedido: 30-6-2018]

[11] "Introducción a XML", *MDN web docs.* [En línea]. Disponible en: https://developer.mozilla.org/es/docs/Introducci%C3%B3n_a_XML [Accedido: 30-6-2018]

[12] “7 razones para programar en Python”, *Blog de empleabilidad y emprendimiento*, 18/09/2016. [En línea]. Disponible en: https://www.bejob.com/7-razones-para-programar-en-python/ [Accedido: 30-6-2018]

[13] “Anaconda Distribution: La Suite más complete para la Ciencia de datos con Python”, *DesdeLinux,* 15/09/2017. [En línea]. Disponible en: https://blog.desdelinux.net/ciencia-de-datos-con-python/ [Accedido: 30-6-2018]

[14] P.Rochina, “Python vs R para el análisis de datos”, *Revista digital INESEM*, 16/11/2016. [En línea]. Disponible en: https://revistadigital.inesem.es/informatica-y-tics/python-r-analisis-datos/ [Accedido: 05-07-2018]

[15] K.Willems, “Choosing R or Python for Data Analysis? An infographic, *Data Camp Community,* 12/07/2015. [En línea]. Disponible en: https://www.datacamp.com/community/tutorials/r-or-python-for-data-analysis#gs.VtgB100 [Accedido: 05-07-2018]

[16] “The ElementTree XML API”, *Python Software Foundation*. [En línea]. Disponible en: https://docs.python.org/3/library/xml.etree.elementtree.html [Accedido: 07-07-2018]

[17] “JSON encoder and decoder”, *Python Software Foundation*. [En línea]. Disponible en: https://docs.python.org/3/library/json.html [Accedido: 15-07-2018]

[18] “Natural Language Toolkit documentation”, *nltk.* [En línea]. Disponible en: http://www.nltk.org/ [Accedido: 15-07-2018]

[19] “API Reference sckit-learn”, *scikit-learn.* [En línea]. Disponible en: http://scikit-learn.org/stable/index.html [Accedido: 15-07-2018]

[20] “¿Qué es JavaScript?”, *MDN web docs.* [En línea]. Disponible en: https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qué_es_JavaScript [Accedido: 17-07-2018]

[21] S.Murray, *Interactive Data Visualization,* 2ª edición. United States of America: O’Reilly, 2013.

[22] “SVG”, *MDN web docs* [En línea]. Disponible en: https://developer.mozilla.org/es/docs/Web/SVG [Accedido: 20-07-2018]

[23] “Data-Driven Documents Tutorial”, *d3js.* [En línea]. Disponible en: https://d3js.org/ [Accedido: 26-07-2018]

[24] M.Bostock, V.Ogievetsky, J.Heer, “D3: Data-Driven Documents”, *Stanford Visualization Group,* 23/10/2011. [En línea]. Disponible en: http://vis.stanford.edu/files/2011-D3-InfoVis.pdf [Accedido: 28-07-2018]

[25] “Crossfilter, Fast multidimensional filtering for coordinated views”, *Crossfilter*. [En línea]. Disponible en: https://github.com/crossfilter/crossfilter [Accedido: 01-08-2018]

[26] “dc.js getting started and how-to guide”, *dc-js.* [En línea]. Disponible en: https://dc-js.github.io/dc.js/ [Accedido: 01-08-2018]

[27] “Interactive charts for browsers and mobile devices.”, *Google Charts*. [En línea]. Disponible en:https://developers.google.com/chart/?hl=es-419 [Accedido: 10-08-2018]

[28] L.Doncel, “La era del algoritmo ha llegado y tus datos son un tesoro”, *El país.* [En línea]. Disponible en: https://elpais.com/economia/2018/03/01/actualidad/1519921981_137226.html [Accedido:13-08-2018]

[29] “Procesamiento de lenguaje natural”, *Instituto de ingeniería del conocimiento.* En línea. Disponible en: http://www.iic.uam.es/soluciones/inteligencia-de-cliente/procesamiento-lenguaje-natural/ [Accedido: 15-08-2018]

[30] “GeoJson”, *GeoJson.* [En línea]. Disponible en: http://geojson.org/ [Accedido: 15-08-2018]

[31] J.Davies, “How the Word Cloud Generator works” [En línea]. Disponible en: https://www.jasondavies.com/wordcloud/about/ [Accedido: 15-08-2018]

[32] J.Brownlee, “A Gentle Introduction to the Bag-of-Words Model”, *Machine Learning Mastery, 09/10/2017.* [En línea]. Disponible en: https://machinelearningmastery.com/gentle-introduction-bag-words-model/ [Accedido: 16-08-2018]

# SUMMARY

## 1. Introduction

The current labour market situation leads to stress or frustration in that moments of searching for a new job. Not only is difficult to find an open-ended job but also a job which suits our requirements. Competition in labour market is increasing so those searching for a job want to do it as efficiently as possible.

To satisfy these needs, job search web portals have been created. In these web sites, a huge amount of job offers and opportunities are uploaded every day. Furthermore, in these sites anyone can filter offers based on several criteria such as category or job location. However, these types of filters are not visual and interactive enough for people to have an easy experience looking for a job. This need to unite the rise of data science and the growth of data visualization tools has led to the creation of a web-based job visualization tool.

The general purpose of this bachelor thesis is the development of a web site with several graphics which allows users to interact with a huge amount of job offers from InfoJobs and Tecnoempleo, two of the most used and known portals for job searches in Spain.

All the objectives of this project are summarized as follows:

- Provide the users with an interactive interface that allows them to analyze job offers from various web portals.
- Present an interface with an adaptable design for correct viewing and use from any browser used.
- Include the possibility of filtering the data from which different graphs are generated in the tool.
- Offer maximum ease of code modification to subsequent developers.
- Carry out the appropriate pre-processing of data for greater significance of the information provided by the graphs.

## 2. State of art

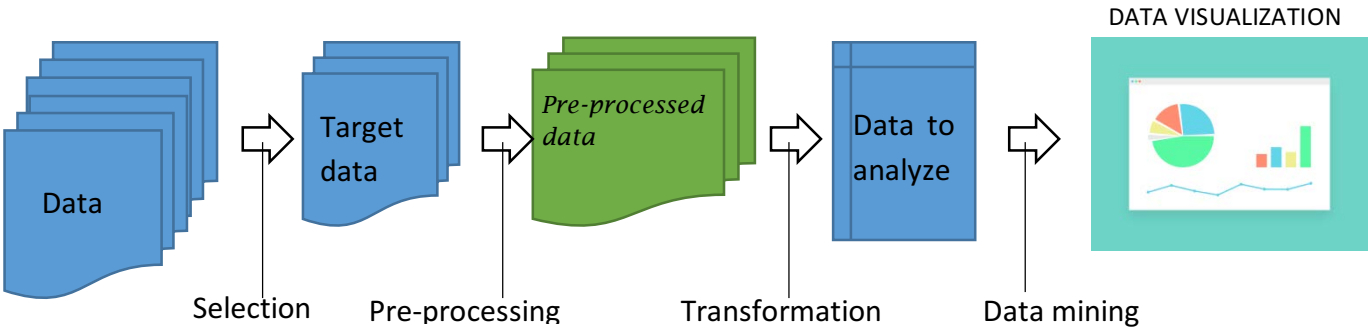
First, it's necessary to explain which data mining and data visualization are all about. After that, they're going to be explained some tools that look like the one created in this project, with their upsides and drawbacks.

### Data mining

Data mining is the process of transforming data in knowledge. Exploration of large amounts of data provides some rules and patterns which are useful to extract information, that's what data mining does. The models to extract these patterns can be predictive or descriptive.

Before data mining process, it is required to pre-process data. In this phase data is manipulated and transformed to be coherent and accessible information.

In Figure 1 it is shown the whole process of Data Mining, including data processing.




This Final Project has followed the above process, from obtaining a large amount of data to the visualization tool that allows the data to be analyzed in an interactive way.

Data visualization

Understanding isolated data is complicated, as it is not very meaningful. In this way, data visualization becomes a very powerful and necessary data interpretation tool for the data to acquire meaning. In addition, people generally understand or receive information better when it appears graphically.

Some of the basic elements in data visualization are: graphs, maps or tables. On the other hand, there are dashboards, compositions of individual visualizations that are interrelated and relatively coordinated. They are very useful for data analysis and decision making.

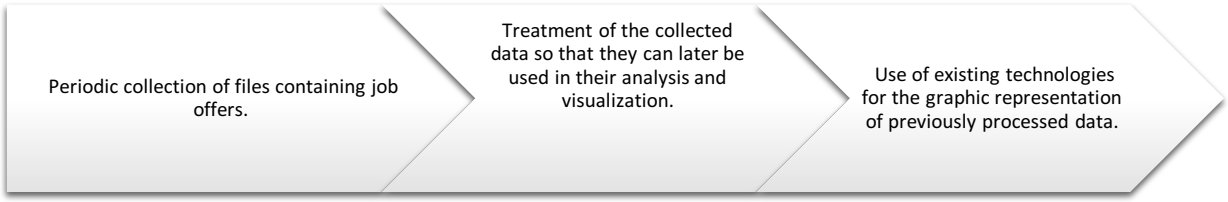
Alternatives to this final project:

- Tecnoempleo.com: It is a web portal where technology companies publish job offers and users can register to be candidates for these jobs. Currently there is a tool on this job portal that allows you to classify and filter out job offers according to technologies, professional functions or countries, among others. Nevertheless, this tool is neither interactive nor attractive to users, which reduces its usefulness. Moreover, the main objective for users of this portal is not to analyze the offers, but to find them.

- InfoJobs: InfoJobs is another employment web portal but, in contrast to the previous one, not only technological jobs but also jobs in various categories (marketing or fine arts). It also allows to filter the offers in the job searches and offers classification by quantity of offers according to several criteria.
- LinkedIn: LinkedIn is a professional social network where companies and professionals promote each other, sharing professional interests to satisfy the professional's search for employment or the recruitment of new employees by companies. This platform offers a "Jobs" section that contains job offers adapted to the interests of each user, however these jobs could not be used in this tool due to use licenses.

## 3. Solution design

The process of creating the job vacancy display tool is divided into three distinct phases: data download, data pre-processing and data visualization.




First, it is necessary to present which technologies have been used in each phase of the project:

Tabla 11-Technologies used in each phase of the Project

| PHASE | TECHNOLOGY |
|---|---|
| **For data collection** | Python |
| **For data pre-processing** | JSON |
| | Python |
| | HTML |
| | CSS |

| | |
|---|---|
| **For data visualization** | JavaScript |
| | SVG |
| | D3.js |

- Python. Python is a programming language that offers many libraries for the treatment of files and large amounts of data, so it was used in this project. To work in python, Jupyter Notebook has been used, which works on the Anaconda distribution platform. The python modules used are:

Tabla 12- Python modules used

| MODULE | USE |
|---|---|
| OS | Allow manipulating the directory structure (to read and write files) |
| LXML | It was used in this project to import the data by reading from a file in xml format. These files are the job offers coming from Tecnoempleo. |
| JSON | This library is used as a JSON encoder and decoder. |
| NLTK | Used to work with text strings or data in natural human language |
| SCIKIT-LEARN | Provides simple and efficient tools for data analysis and data mining |

- HTML (HyperText Markup Language): Defines a basic structure or DOM and code for defining the content of a web page.
- CSS: It is a style language that defines the presentation of documents written in HTML, regarding fonts, colors, margins, heights and widths, etc.
- JavaScript: JavaScript allows you to create dynamic web pages by manipulating the DOM after the page has been loaded into the browser. It is these dynamic properties that have caused this language to be used for programming the job vacancy analysis tool.
- SVG (Scalable vector graphics): SVG is a text-based image format. It provides several facilities that make generating and manipulating images more consistent and faster than doing so with HTML.
- D3.js: It is a JavaScript library for data visualization that uses other technologies such as HTML, SVG and CSS. It allows direct manipulation of the DOM by

associating data with elements of the DOM and using dynamic transformations to generate or modify the content of those elements.

To be able to work with large amounts of data and create a tool that allows interaction with them, other tools built on d3 were used such as Crossfilter and DC.js.

**Crossfilter** is a JavaScript library which allows you to work with a complex and large data set quickly, so that you can easily filter or calculate sums within this set. This allows you to calculate the number of job offers that meet a certain criterion.

**DC.js** is an optimized library for large data sets. The aim of this library is to quickly and easily display the results of the cross-filtering (Crossfilter) using different types of dynamic graphics

## 4. Developing the solution

### 4.1 Downloading data

It consists of downloading data, job offers from the databases of the employment web portals Infojobs and Tecnoempleo. For the download of each platform different scripts written in python are used. The data is downloaded daily.

In the case of InfoJobs, these data are stored in a file written in JSON format and, in addition, these files are stored in different folders depending on their category: computer and telecommunications (IFC), marketing and communication (MC) or Design and graphic arts (ARG).

On the other hand, in order to store the offers of employment of Tecnoempleo, the content of the consulted web page is saved in a file in format xml.

### 4.2 Pre-processing data

In the digital age people are generating more and more data. However, these data have no informative value if they are not processed beforehand.

In this phase of the project, the main objective was to generate a JSON format file containing all the job offers with certain fields that define them. Afterwards, that JSON file was the one used to generate the different graphs in the next phase of the project.

Data pre-processing includes several consecutive phases which are shown in the following table:

Tabla 13-Parts of data pre-processing phase

| Phases | Definition |
|---|---|
| **Data collection** | It consists of downloading the data previously explained |
| **Data cleansing** | Some temporary files that were automatically stored in the download had to be deleted. |
| **Data transformation** | It consists of the homogenization of the data of all the offers, independently of their origin (Tecnoempleo or Infojobs).<br>To do this, it was necessary to fill in those attributes that were incomplete or inconsistent in some offers. Furthermore, it was necessary to discretize some data, such as salary or years of experience. |
| **Data reduction** | It is the selection of those attributes that provide us with useful information for the graphs. |

Those necessary attributes for the generation of the graphs were:

- Creation date (DD/MM/YYYY)
- City: where the job is offered.
- Experience: years of experience required as a minimum.
- Category: It can be informatics and telecommunications, marketing and communication or design and graphic arts.
- Country: where the job is offered.
- Type of contract: temporary, open-ended, etc.
- Working hours.
- The number of vacancies.
- Minimum requirements: programming languages, teamwork, languages, etc.
- Salary.
- Job offer description.
- Source: InfoJobs or Tecnoempleo.

In the data pre-processing phase, it was necessary to process unstructured data, “Minimum requirements”, particularly. In this case, it was used the technique called **natural language processing**, to process each word of the text of "minimum requirements" and thus be able to have a count of the times that each word appeared in a text. This count was necessary because we wanted to make a **bag of words** and then generate a graph called **word cloud**.

In this task, we used the library of python *scikit-learn*, used to obtain word counts, and a lemmatizer. A lemmatizer normalizes words that appear in different forms (plural, verb tenses or languages) to a single common form. Thanks to the use of the lemmatizer it is possible to reduce considerably the unnecessary or redundant words when creating the vocabulary of words to count.

### 4.3 Data visualization

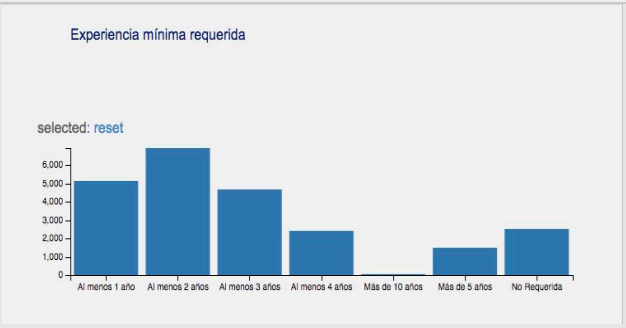
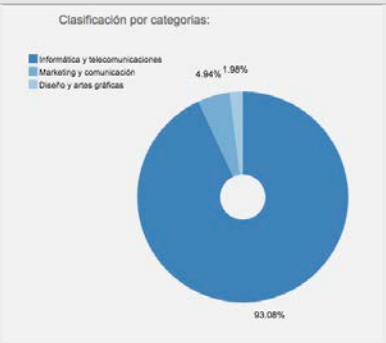
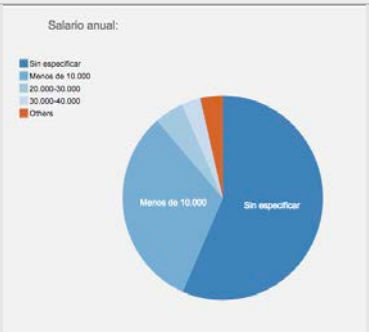
The main tools that were used to create the graphs or charts from the input data (job offers with their attributes) are: dc.js and Crossfilter, both built on d3.js library.

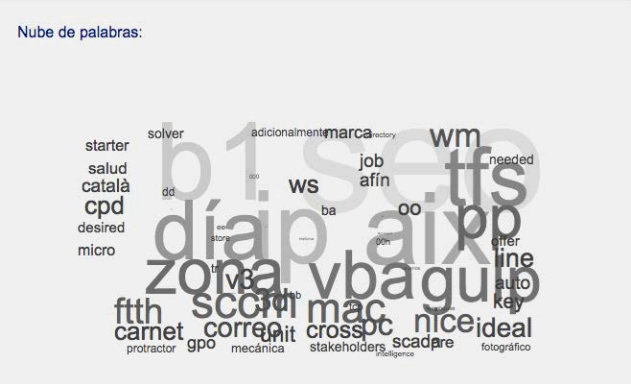
These tools allow graphical representations of large datasets and fast interaction between them. Particularly, for each graph you have to define the variable or attribute that you want to represent (the dimension) and how you want to represent, for example, its total sum, that is: the group.

The input to this last phase is a JSON format file containing: the data (processed job offers), geoJson (coordinates of the map of Spain) and a vocabulary of words necessary to elaborate the WordCloud mentioned previously.

Thanks to these data, the following charts were constructed:

Tabla 14-Attributes and charts

| Displayed Data | Chart type | |
|---|---|---|
| **Data source** | Pie chart |  |
| **Minimum experience** | Bar chart |  |
| **Category** | Pie chart |  |
| **Salary** | Pie chart |  |

| | | |
|---|---|---|
| **Geographical distribution of the offers** | Map |  |
| **Minimum requirements** | Word Cloud |  |
| **Creation date** | Bar chart<br>+<br>Line Char |  |

## 5. Planning and budget

The development of the present project took 8 months in total. During this time, three main phases can be distinguished:

- Start of the project (training)
- Visualization tool development
- Memory, evaluation and improvements.

It is also necessary to have a budget that accounts for expenditures on manpower, software and hardware.

Regarding software, no expense is considered since educational licenses and open and open source platforms were used.

Regarding hardware, only the use of a computer is considered, whose useful life is 5 years, with a cost of 1000€. We conclude that the total investment in hardware was 133,33€.

Finally, the cost of manpower was:

Tabla 15- Division of cost of manpower

| Person | Category | Hours | €/h | Total |
|---|---|---|---|---|
| **Ángela Moreno Martínez** | Newly qualified | 400 h | 23,43€/h | 9372€ |
| **Vanessa Gómez Verdejo** | PhD | 40 h | 35,33€/h | 1413,2€ |
| | | | **Cost of the manpower:** | **10785,20 €** |

Considering the costs of hardware as well as manpower, the complete approximate budget would be a total of 10918,53€.

## 6. Conclusion and future work

The web tool for displaying job offers provides a comfortable and simple interface for professionals to be aware of which are the skills required, as well as the jobs offered by companies recognized in various sectors.

The opinions of the surveys have been considered in order to propose new future lines of work.

It is considered possible that the web portals, from which the job offers used in this project are obtained, InfoJobs and Tecnoempleo, can use this web tool to provide more content and information to users, as well as ease in the analysis of the offers (socio-economic impact).

Possible improvements:

- To have the data stored in organized databases, so that these tables can be accessed through queries from the visualization tool.
- The use of other job vacancy databases for greater variability and significance of the data.
- Use of 3D graphics.
- Analysis of topics.
- Development of a mobile application.

## ANEXO

# Encuesta herramienta de análisis

A continuación, se presentan una serie de preguntas que pertenecen a una encuesta sobre la herramienta web de visualización de ofertas de empleo que debe haber visitado previamente. La participación en esta encuesta ayuda en el proceso de un Trabajo de Fin de Grado.

*Obligatorio

1) Valore globalmente la herramienta *

| | 1 | 2 | 3 | 4 | 5 | |
|---|---|---|---|---|---|---|
| Muy en desacuerdo | ○ | ○ | ○ | ○ | ○ | Muy de acuerdo |

2) Valore la interfaz de la página web (diseño y facilidad de uso) *

| | 1 | 2 | 3 | 4 | 5 | |
|---|---|---|---|---|---|---|
| Muy en desacuerdo | ○ | ○ | ○ | ○ | ○ | Muy de acuerdo |

3) Valore la utilidad de la herramienta *

| | 1 | 2 | 3 | 4 | 5 | |
|---|---|---|---|---|---|---|
| Muy en desacuerdo | ○ | ○ | ○ | ○ | ○ | Muy de acuerdo |

4) Seleccione el uso que le daría a la presente herramienta web *

- ○ Ayuda en la búsqueda de empleo
- ○ Análisis de ofertas de empleo
- ○ Otro: _____

5) Introduzca cuál(es) de todos los gráficos le resulta más util(es) *

Elige

¿Usarías esta herramienta como complemento en la búsqueda de empleo? *

- ○ Sí
- ○ No
- ○ Tal vez

¿Cómo mejorarías la herramienta? *

Tu respuesta